Hewlett-Packard Company

TPC Benchmark™ H
Full Disclosure Report

HP ProLiant DL585 Cluster 48P
using
Oracle Database 10g Enterprise Edition with
Real Application Cluster and
Partitioning; and
Red Hat Enterprise Linux AS 3

Second Edition
October 2004

© 2004 Hewlett Packard Company. All rights reserved

Second Edition – October 2004

Hewlett Packard Company, the Sponsor of this benchmark test, believes that the information in this document is accurate as of the publication date. The information in this document is subject to change without notice. The Sponsor assumes no responsibility for any errors that may appear in this document.

The pricing information in this document is believed to accurately reflect the current prices as of the publication date. However, the Sponsor provides no warranty of the pricing information in this document.

Benchmark results are highly dependent upon workload, specific application requirements, and system design and implementation. Relative system performance will vary as a result of these and other factors. Therefore, the TPC Benchmark H should not be used as a substitute for a specific customer application benchmark when critical capacity planning and/or product evaluation decisions are contemplated.

All performance data contained in this report was obtained in a rigorously controlled environment. Results obtained in other operating environments may vary significantly. No warranty of system performance or price/performance is expressed or implied in this report.

© Hewlett Packard Company 2004.

All rights reserved. Permission is hereby granted to reproduce this document in whole or in part provided the copyright notice printed above is set forth in full text or on the title page of each item reproduced.

HP ProLiant is a registered trademark of Hewlett Packard Company.

Oracle Database 10g Enterprise Edition with Real Application Cluster and Partitioning are trademarks of Oracle Corporation

TPC Benchmark, TPC-H, QppH, QthH and QphH are trademarks of the Transaction Processing Performance Council.

All other brand or product names mentioned herein must be considered trademarks or registered trademarks of their respective owners.

Abstract

Overview

This report documents the methodology and results of the TPC Benchmark™ H test conducted on the HP ProLiant DL585 Cluster 48Pusing Oracle Database 10g Enterprise Edition with Real Application Cluster and Partitioning, in conformance with the requirements of the TPC Benchmark™ H Standard Specification, Revision 2.1.0. The operating system used for the benchmark was Red Hat Enterprise Linux AS 3.

The benchmark results are summarized in the following table.

Hardware	Software	Total System Cost	QppH @ 1TB	QthH @ 1TB	QphH @ 1TB	$ / QphH @ 1TB
HP ProLiant DL585 Cluster 48P	Oracle Database 10g Enterprise Edition with Real Application Cluster and Partitioning, and Red Hat Enterprise Linux AS 3	$2,106,123	46463.6	26577.6	235141.0	$60

The TPC Benchmark™ H was developed by the Transaction Processing Performance Council (TPC). The TPC was founded to define transaction processing benchmarks and to disseminate objective, verifiable performance data to the industry.

Standard and Executive Summary Statements

Executive Summary and Numerical Quantities Summary of the benchmark results for the HP ProLiant DL585 Cluster 48Pcan be found in the following pages.

Auditor

The benchmark configuration, environment and methodology were audited by Lorna Livingtree of Performance Metrics Inc. to verify compliance with the relevant TPC specifications.

© 2004 Hewlett Packard Company. All rights reserved

Table Of Contents

ABSTRACT 3
OVERVIEW 3
STANDARD AND EXECUTIVE SUMMARY STATEMENTS 3
AUDITOR 3

TABLE OF CONTENTS IV

1.0 GENERAL ITEMS 6
1.1 TEST SPONSOR 6
1.2 PARAMETER SETTINGS 6
1.3 CONFIGURATION ITEMS 7

2.0 CLAUSE 1: LOGICAL DATABASE DESIGN 9
2.1 DATABASE DEFINITION STATEMENTS 9
2.2 PHYSICAL ORGANIZATION OF DATABASE 9
2.3 HORIZONTAL PARTITIONING 9
2.4 REPLICATION 9

3.0 CLAUSE 2: QUERIES AND REFRESH FUNCTIONS RELATED ITEMS 10
3.1 QUERY LANGUAGE 10
3.2 RANDOM NUMBER GENERATION 10
3.3 SUBSTITUTION PARAMETERS GENERATION 10
3.4 QUERY TEXT AND OUTPUT DATA FROM DATABASE 10
3.5 QUERY SUBSTITUTION PARAMETERS AND SEEDS USED 10
3.6 ISOLATION LEVEL 10
3.7 REFRESH FUNCTIONS 10

4.0 CLAUSE 3: DATABASE SYSTEM PROPERTIES 12
4.1 ATOMICITY REQUIREMENTS 12
4.2 CONSISTENCY REQUIREMENTS 12
4.3 ISOLATION REQUIREMENTS 13
4.4 DURABILITY REQUIREMENTS 14

5.0 CLAUSE 4: SCALING AND DATABASE POPULATION 16
5.1 INITIAL CARDINALITY OF TABLES 16
5.2 DISTRIBUTION OF TABLES AND LOGS ACROSS MEDIA 16
5.3 MAPPING OF DATABASE PARTITIONS/REPLICATIONS 18
5.4 IMPLEMENTATION OF RAID 19
5.5 DBGEN MODIFICATIONS 19
5.6 DATABASE LOAD TIME 19
5.7 DATA STORAGE RATIO 19
5.8 DATABASE LOAD MECHANISM DETAILS AND ILLUSTRATION 20
5.9 QUALIFICATION DATABASE CONFIGURATION 20

6.0 CLAUSE 5: PERFORMANCE METRICS AND EXECUTION RULES RELATED ITEMS 21
6.1 STEPS IN THE POWER TEST 21
6.2 TIMING INTERVALS FOR EACH QUERY AND REFRESH FUNCTION 21
6.3 NUMBER OF STREAMS FOR THE THROUGHPUT TEST 21
6.4 START AND END DATE/TIMES FOR EACH QUERY STREAM 21
6.5 TOTAL ELAPSED TIME FOR THE MEASUREMENT INTERVAL 21
6.6 REFRESH FUNCTION START DATE/TIME AND FINISH DATE/TIME 21
6.7 TIMING INTERVALS FOR EACH QUERY AND EACH REFRESH FUNCTION FOR EACH STREAM 21

© 2004 Hewlett Packard Company. All rights reserved

6.8 PERFORMANCE METRICS 22
6.9 THE PERFORMANCE METRIC AND NUMERICAL QUANTITIES FROM BOTH RUNS 22
6.11 SYSTEM ACTIVITY BETWEEN TESTS 22

7.0 CLAUSE 6: SUT AND DRIVER IMPLEMENTATION RELATED ITEMS 23
7.1 DRIVER 23
7.2 IMPLEMENTATION SPECIFIC LAYER (ISL) 23
7.3 PROFILE-DIRECTED OPTIMIZATION 23

8.0 CLAUSE 7: PRICING RELATED ITEMS 24
8.1 HARDWARE AND SOFTWARE USED 24
8.2 TOTAL 3 YEAR PRICE 24
8.3 AVAILABILITY DATE 24
8.4 COUNTRY-SPECIFIC PRICING 24

9.0 CLAUSE 9: RELATED ITEMS 25
9.1 AUDITORS' REPORT 25

APPENDIX A: TUNABLE PARAMETERS 26

APPENDIX B: DATABASE BUILD SCRIPTS 29

APPENDIX C: ACID SCRIPTS 61

APPENDIX D: QUERY TEXT AND OUTPUT 86

APPENDIX E: SEED AND INPUT PARAMETERS 101

APPENDIX F: BENCHMARK SCRIPTS 104

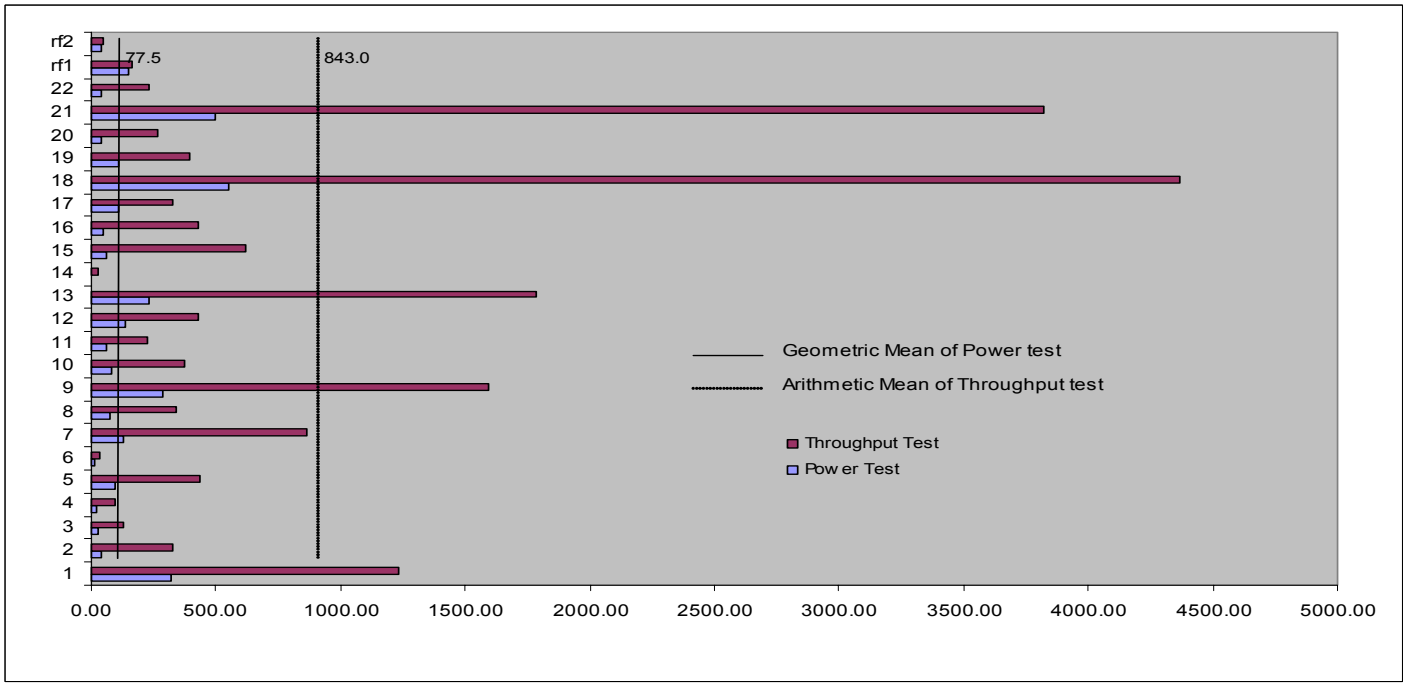
APPENDIX G: PRICE QUOTES 120

© 2004 Hewlett Packard Company. All rights reserved

invent	**HP ProLiant DL585 Cluster 48P**	TPC-H Rev. 2.1.0 Report Date: **October 22, 2004**

Total System Cost	Composite Query per Hour Metric	Price / Performance
$2,106,123	**35141.0** QphH@1000GB	**$60** $ / QphH@1000GB

Database Size	Database Manager	Operating System	Other Software	Availability Date
1000GB	**Oracle Database 10g Enterprise Edition with Real Application Clusters and Partitioning**	**Red Hat Enterprise Linux AS 3**		**October 21, 2004**



Database Load Time = 4:12:23	Load Included Backup: Y	Total Data Storage / Database Size = 14.69
RAID (Base tables only): N	RAID (Base tables and auxiliary data structures): N	RAID (All): N

12 Node Cluster Configuration :

Processors (per node) :	4 x 2.2GHz/1MB AMD Opteron Model 848
Memory (per node) :	8 GB
OS Disk Drives (per node) :	2 x 36GB 15krpm HDD Ultra320
NICs (per node) :	2 x on-board
Cluster interconnect (per node) :	1 x InfiniCon Systems InfiniServ 7000 HCA (interconnected using InfiniCon Systems InfinIO 3016 switch)
Disk Controllers (per node) :	6 x hp StorageWorks fca 2214DC
Storage Area Network :	12 x hp StorageWorks SAN Switch 2/16 48 x hp StorageWorks MSA1000 408 x 36GB 15krpm HDD Ultra320
Total Storage :	14688GB

© 2004 Hewlett Packard Company. All rights reserved.

	HP ProLiant DL585 Cluster 48P				TPC-H Rev. 2.1.0		
					Report Date:		22-Oct-04
Description	**Part Number**	**Brand**	**Pricing**	**Unit Price**	**Qty**	**Extended Price**	**3 yr. Maint. Price**
Server Hardware							
ProLiant DL585 AMD Opteron™ 2.2GHz/ 1MB, 2GB - 2P Model	356820-B21		1	11,999	12	143,988	
Two AMD 848 Opteron™ 2.2GHz/1MB processors	359708-2PR		1	9,398	12	112,776	
1GB of Advanced ECC PC2100 DDR SDRAM DIMM memory kit	300679-B21		1	459	72	33,048	
36GB, 15krpm HDD Ultra320 HP	286776-B22		1	349	24	8,376	
S5500 15" Monitor	261602-001		1	129	12	1,548	
Scroll Mouse-Carbon	231947-B21		1	5	12	60	
PS/2 Easy Access Internet Keyboard	265977-001		1	12	12	144	
hp StorageWorks fca 2214DC 2Gb, 64-bit/133Mhz PCI-X FC HBA	321835-B21		1	2,500	72	180,000	
Servers 500 Series CarePaq, 3yr 7x24, 4hr resp.	401782-002		1	1,795	12		21,540
Rack Model 9142 (42U - Opal) - Flat Pallet	120663-B21		1	1,321	9	11,889	
Rack Sidewall Kit	120670-B21		1	207	1	207	
UPS T1500 XR	204155-002		1	621	1	621	
hp ProCurve Switch 4148gl	J4888A		1	1,934	2	3,868	
hp ProCurve Switch gl 100/1000-T module	J4863A		1	812	4	3,248	
hp ProCurve Switch gl 100/1000-T module Support 3YR 24X7 4HR	U2856E		1	1,080	2		2,160
InfiniCon InfiniIO 3016 - 32 Port infiniBand Switch (Inc. 2 spares, 2 Addi. Yrs Warranty)	3016-SW1	InfiniCon	4	8,769	3	26,307	1,830
InfiniCon InfiniServ 7000 Low Profile HCA 128 MB Std. (Inc. 2 spares, 2 Addi. Yrs Warrant	7104-HCA-128LP	InfiniCon	4	707	14	9,898	900
InfiniCon Systems InfiniView embedded subnet manager	SW79-SM-V2K	InfiniCon	4	1,000	1	1,000	1,050
InfiniCon 4x-4x cable 5m (Inc. 2 spares)	7104-5M-CABLE	InfiniCon	4	128	14	1,792	
					Subtotal	**538,770**	**27,480**
Storage							
hp Storageworks Modular SAN Array 1000	201723-B22		1	9,995	48	479,760	
hp StorageWorks Modular SAN Array 1000 Support 3YR 24x7 4HR	402164-002		1	3,538	48		169,824
hp StorageWorks SAN switch 2/16-EL	322120-B21		1	6,400	12	76,800	
hp SAN Switch 2/32 Support 3YR 24X7 4HR	161299-002		1	3,399	12	40,788	
5m LC to LC Cable Kit	221692-B22		1	82	96	7,872	
15m LC to LC Cable Kit	221692-B23		1	103	96	9,888	
2Gb SFF-SW Trncvr Kit	221470-B21		1	199	192	38,208	
2Gb SFF-SW Trncvr Kit (10% spares)	221470-B21		1	199	20	3,980	
36GB, 15krpm HDD Ultra320 HP	286776-B22		1	349	408	142,392	
36GB, 15krpm HDD Ultra320 HP (10% spares)	286776-B22		1	349	41	14,309	
					Subtotal	**813,997**	**169,824**
Software							
Oracle Database 10g Enterprise Edition Named User Plus for 3 years	run-time	Oracle	2	10,000	48	480,000	
Real Application Clusters Named User Plus for 3 Yrs	run-time	Oracle	2	5,000	48	240,000	
Partitioning Named User Plus for 3 Yrs	run-time	Oracle	2	2,500	48	120,000	
Database Server Support Package (12 servers, 3 Yrs)	run-time	Oracle	2	24,000	3		72,000
Red Hat Enterprise Linux AS 3.0 (Premium Edition for x86)	SVC0084	RedHat	3	2,499	12	29,988	
2 Addi. Yrs Subs. to Red Hat Enterprise Linux AS 3.0 (Premium Edition x86)	SVC0084	RedHat	3	2,499	24		59,976
					Subtotal	**869,988**	**131,976**
*hp Large Purchase discount			1			(229,970)	(33,542)
Oracle Mandatory E-Business Discount			2			(182,400)	
					Total	**1,810,385**	**295,738**
						3-Year Cost of Ownership:	**$2,106,123**
						QphH Rating:	**35141.0**
Price Key: 1-HP at 17% discount, 2-Oracle, 3-Red Hat, 4- InfiniCon Systems, Oracle pricing contact: Mary Beth Pierantoni, mary.beth.pierantoni@oracle.com, 916-315-5081. *All discounts are based on US list prices and for similar quantities and configurations.							
					$ / QphH@1000GB:		**$60**

Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are not permitted. Special prices based on assumptions about past or future purchases are not permitted. All discounts reflect standard pricing policies for the listed components. For complete details, see the pricing sections of the TPC benchmark specifications. If you find that the stated prices are not available according to these terms, please inform the TPC at pricing@tpc.org. Results independently audited by Lorna Livingtree of Performance Metrics Inc. Thank you.

© 2004 Hewlett Packard Company. All rights reserved.

Numerical Quantities

Measurement Results:

Database Scale Factor	= 1000
Total Data Storage / Database Size	= 14.69
Start of Database Load	= 7/7/2004 09:36:46
End of Database Load	= 7/7/2004 13:05:29
Database Backup Elapse Time	= 00:43:40
Database Load Time + Backup	= 04:12:23
Query Streams for Throughput Test	= 7
TPC-H Power	= 46463.6
TPC-H Throughput	= 26577.6
TPC-H Composite Query-per-Hour Metric (QphH@1000GB)	= 35141.0
Total System Price Over 3 Years	= $2,106,123
TPC-H Price/ Performance Metric ($/QphH@1000GB)	= $60

Measurement Intervals:

Measurement Interval in Throughput Test (Ts)	= 20859.7 seconds

Duration of Stream Execution:

Stream ID	Seed	Start Date	Start Time	End Date	End Time	Duration
Stream00	707130529	7/7/2004	14:11:35	7/7/2004	15:01:10	0:49:35
Stream01	707130530	7/7/2004	15:01:53	7/7/2004	20:05:51	5:03:58
Stream02	707130531	7/7/2004	15:01:53	7/7/2004	20:24:51	5:22:58
Stream03	707130532	7/7/2004	15:01:53	7/7/2004	19:54:56	4:53:03
Stream04	707130533	7/7/2004	15:01:53	7/7/2004	19:56:57	4:55:04
Stream05	707130534	7/7/2004	15:01:53	7/7/2004	20:19:21	5:17:28
Stream06	707130535	7/7/2004	15:01:53	7/7/2004	20:22:15	5:20:22
Stream07	707130536	7/7/2004	15:01:54	7/7/2004	19:48:07	4:46:13
Refresh		7/7/2004	20:27:33	7/7/2004	20:49:32	0:21:59

© 2004 Hewlett Packard Company. All rights reserved.

TPC-H Timing Intervals (in seconds)

Query	Stream00	Stream01	Stream02	Stream03	Stream04	Stream05	Stream06	Stream07	Min Qi	Max Qi	Avg Qi
1	317.48	407.67	1785.96	1003.33	1370.17	1192.35	1791.94	1079.10	407.67	1791.94	1232.93
2	41.28	345.15	377.29	332.84	317.87	273.16	300.48	336.57	273.16	377.29	326.19
3	24.58	122.09	183.07	100.31	38.00	177.98	137.73	131.89	38.00	183.07	127.30
4	17.53	30.50	37.97	91.46	143.71	113.00	124.72	102.50	30.50	143.71	91.98
5	92.06	554.51	524.37	466.58	301.84	339.08	355.00	486.95	301.84	554.51	432.62
6	12.87	19.23	21.55	62.40	22.61	22.65	16.71	70.67	16.71	70.67	33.69
7	132.66	747.32	968.54	836.97	797.13	772.12	801.19	1119.27	747.32	1119.27	863.22
8	71.78	320.90	282.72	559.59	292.75	225.33	391.94	304.55	225.33	559.59	339.68
9	287.59	1628.92	1775.06	1732.22	1626.55	384.70	1620.72	2389.82	384.70	2389.82	1594.00
10	78.69	467.46	403.19	409.39	389.17	125.74	365.93	439.84	125.74	467.46	371.53
11	62.68	234.64	265.42	201.01	229.30	216.28	225.03	190.43	201.01	265.42	223.16
12	133.26	303.54	370.21	509.00	487.14	210.44	574.11	542.51	210.44	574.11	428.14
13	234.69	1788.25	1997.18	1588.60	2072.81	1972.64	1464.37	1594.83	1464.37	2072.81	1782.67
14	8.32	30.29	20.02	19.59	18.93	22.02	39.60	62.06	18.93	62.06	30.36
15	58.88	593	513.5	634.8	740.3	793.7	483.3	575.1	483.30	793.70	619.10
16	49.50	541.16	338.17	357.05	369.57	353.23	664.62	371.35	338.17	664.62	427.88
17	106.77	200.18	647.49	409.52	215.11	123.13	528.47	173.01	123.13	647.49	328.13
18	550.85	4229.24	4830.85	4423.50	4015.34	5060.44	4768.80	3245.52	4015.34	5060.44	4367.67
19	112.17	340.03	298.88	330.57	286.32	659.42	266.81	602.42	266.81	659.42	397.78
20	38.77	747.28	184.34	228.59	57.14	250.78	201.66	170.32	57.14	747.28	262.87
21	499.63	4513.47	3299.78	3010.88	3725.67	5415.59	3815.86	2950.76	3010.88	5415.59	3818.86
22	42.54	72.39	251.72	274.62	185.67	343.76	282.74	233.76	72.39	343.76	234.95
rf1	149.35	162.3	182.7	166.8	161.7	153.6	156.4	161.3	153.6	182.7	163.5
rf2	39.38	48.8	58.7	39.8	43.3	54.5	50.4	41.2	39.8	58.7	48.1

© 2004 Hewlett Packard Company. All rights reserved.

1.0 General Items

1.1 Test Sponsor

A statement identifying the benchmark sponsor(s) and other participating companies must be provided.

Hewlett Packard Company sponsored this benchmark. The benchmark was developed and engineered by Hewlett Packard Company and Oracle Corporation. Testing took place at HP Database Performance Engineering Laboratory in Houston, Texas.

1.2 Parameter Settings

Settings must be provided for all customer-tunable parameters and options which have been changed from the defaults found in actual products, including by not limited to:

- *Database Tuning Options*
- *Optimizer/Query execution options*
- *Query processing tool/language configuration parameters*
- *Recovery/commit options*
- *Consistency/locking options*
- *Operating system and configuration parameters*
- *Configuration parameters and options for any other software component incorporated into the pricing structure*
- *Compiler optimization options*

This requirement can be satisfied by providing a full list of all parameters and options, as long as all those which have been modified from their default values have been clearly identified and these parameters and options are only set once.

Appendix A contains Database and Operating system configuration and tunable parameters.

© 2004 Hewlett Packard Company. All rights reserved

1.3 Configuration Items

Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences. This includes, but is not limited to:

- *Number and type of processors*
- *Size of allocated memory, and any specific mapping/partitioning of memory unique to the test.*
- *Number and type of disk units (and controllers, if applicable).*
- *Number of channels or bus connections to disk units, including their protocol type.*
- *Number of LAN (e.g. Ethernet) Connections, including routers, workstations, terminals, etc., that were physically used in the test or are incorporated into the pricing structure.*
- *Type and the run-time execution location of software components (e.g., DBMS, query processing tools/languages, middle-ware components, software drivers, etc.).*

The HP ProLiant DL585 Cluster 48P is depicted in Figure 1.1 consisted of:

Processors (per node) :	4 x 2.2GHz/1MB AMD Opteron Model 848
Memory (per node) :	8 GB
OS Disk Drives (per node) :	2 x 36GB 15krpm HDD Ultra320
NICs (per node) :	2 x on-board
Cluster interconnect (per node) :	1 x InfiniServ 7000 (connected to InfinIO 3016 switch)
Disk Controllers (per node) :	6 x hp StorageWorks fca 2214DC
Storage Area Network :	12 x hp StorageWorks SAN Switch 2/16
	48 x hp StorageWorks MSA1000
	408 x 36GB 15krpm HDD Ultra320
Total Storage :	14688GB

The storage area network (SAN) consisted of 12 hp SAN Switch 2/16s and 48 hp StorageWorks MSA1000s. The hp ProLiant DL585 servers had six hp StorageWorks fca 2214DC (dual port) HBAs; each port connected to one of twelve hp SAN Switch 2/16s. Each hp SAN Switch 2/16s had four hp StorageWorks MSA1000s connected to it.

Each hp StorageWorks MSA1000 had two RAID0 volumes of four 36GB 15krpm HDD Ultra320 each for data; 12 MSA 1000s had one additional RAID1+0 volume of two 36GB 15krpm HDD Ultra320 each for redo logs. The array accelerator cache was set to 80% read 20% write for data volumes and disabled for redo log volumes.

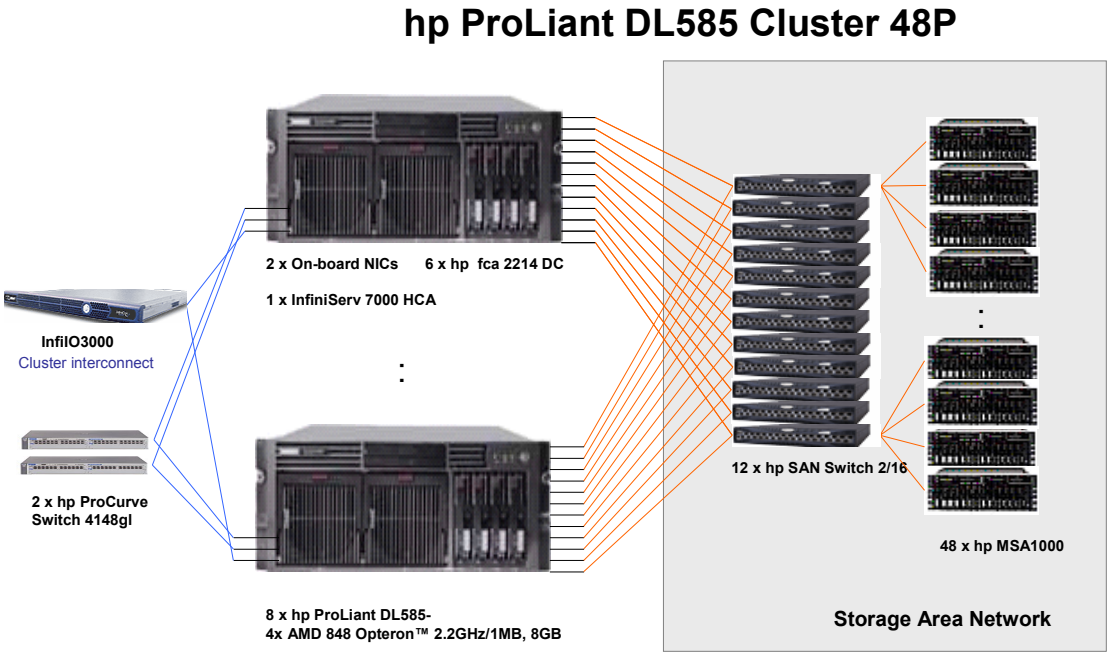
All tables except NATION and REGION had 96 horizontal partitions. The temp files and flat files were also distributed across the 96 volumes, two per hp StorageWorks MSA1000s.

One InfiniCon InfinIO 3016 switch (Infiniband) switch was used as cluster interconnect.

A detailed description of distribution of database files can be found in Table 5.2.

© 2004 Hewlett Packard Company. All rights reserved

Figure 1.1: Benchmarked and Priced Configuration



© 2004 Hewlett Packard Company. All rights reserved

2.0 Clause 1: Logical Database Design

2.1 Database Definition Statements

Listings must be provided for all table definition statements and all other statements used to set up the test and qualification databases. (8.1.2.1)

Appendix B contains the database build scripts.

2.2 Physical Organization of Database

The physical organization of tables and indices, within the test and qualification databases, must be disclosed. If the column ordering of any table is different from that specified in Clause 1.4, it must be noted.

Please refer Appendix B for column reordering of tables.

2.3 Horizontal Partitioning

Horizontal partitioning of tables and rows in the test and qualification databases (see Clause 1.5.4) must be disclosed.

Horizontal partitioning was used for all tables except NATION and REGION as described in Appendix B.

2.4 Replication

Any replication of physical objects must be disclosed and must conform to the requirements of Clause 1.5.6.

The database was not replicated.

© 2004 Hewlett Packard Company. All rights reserved

3.0 Clause 2: Queries and Refresh Functions Related Items

3.1 Query Language

The query language used to implement the queries must be identified.

SQL was the query language used to implement all queries.

3.2 Random Number Generation

The method of verification for the random number generation must be described unless the supplied DBGEN and QGEN were used.

TPC supplied versions 1.3.0 of DBGEN and QGEN were used for this TPC-H benchmark.

3.3 Substitution Parameters Generation

The method used to generate values for substitution parameters must be disclosed. If QGEN is not used for this purpose, then the source code of any non-commercial tool used must be disclosed. If QGEN is used, the version number, release number, modification number and patch level of QGEN must be disclosed.

The supplied QGEN version 1.3.0 was used to generate the substitution parameters.

3.4 Query Text and Output Data from Database

The executable query text used for query validation must be disclosed along with the corresponding output data generated during the execution of the query text against the qualification database. If minor modifications (see Clause 2.2.3) have been applied to any functional query definitions or approved variants in order to obtain executable query text, these modifications must be disclosed and justified. The justification for a particular minor query modification can apply collectively to all queries for which it has been used. The output data for the power and throughput tests must be made available electronically upon request..

Appendix D contains the query text and output.

3.5 Query Substitution Parameters and Seeds Used

All the query substitution parameters used during the performance test must be disclosed in tabular format, along with the seeds used to generate these parameters.

Appendix E contains the query substitution parameters and seed used.

3.6 Isolation Level

The isolation level used to run the queries must be disclosed. If the isolation level does not map closely to one of the isolation levels defined in Clause 3.4, additional descriptive detail must be provided.

The queries and transactions were run with the isolation level set to “Level 3” (repeatable read).

3.7 Refresh Functions

The details of how the refresh functions were implemented must be disclosed (including source code of any non-commercial program used).

© 2004 Hewlett Packard Company. All rights reserved

The refresh function is part of the implementation-specific layer/driver code included in Appendix F.

© 2004 Hewlett Packard Company. All rights reserved

4.0 Clause 3: Database System Properties

4.1 Atomicity Requirements

The results of the ACID tests must be disclosed along with a description of how the ACID requirements were met. This includes disclosing the code written to implement the ACID Transaction and Query.

Appendix C contains the source code for the ACID transactions.

4.1.1 Atomicity of the Completed Transactions

Perform the ACID Transaction for a randomly selected set of input data and verify that the appropriate rows have been changed in the ORDER, LINEITEM, and HISTORY tables.

The following steps were performed to verify the Atomicity of the completed ACID transactions:

1. The total price from the ORDERS table and the extended price from the LINEITEM table were retrieved for a randomly selected order key.
2. The ACID Transaction was performed using the order key from step 1.
3. The ACID Transaction committed.
4. The total price from the ORDERS table and the extended price from the LINEITEM table were retrieved for the same order key. It was verified that the appropriate rows had been changed.

4.1.2 Atomicity of Aborted Transactions

Perform the ACID transaction for a randomly selected set of input data, submitting a ROLLBACK of the transaction for the COMMIT of the transaction. Verify that the appropriate rows have not been changed in the ORDER, LINEITEM, and HISTORY tables.

The following steps were performed to verify the Atomicity of the aborted ACID transactions:

1. The total price from the ORDERS table and the extended price from the LINEITEM table were retrieved for a randomly selected order key.
2. The ACID Transaction was performed using the order key from step 1. The transaction was stopped prior to the commit.
3. The ACID Transaction was ROLLED BACK.
4. The total price from the ORDERS table and the extended price from the LINEITEM table were retrieved for the same order key. It was verified that the appropriate rows had not been changed.

4.2 Consistency Requirements

Consistency is the property of the application that requires any execution of transactions to take the database from one consistent state to another.

A consistent state for the TPC-H database is defined to exist when:

*O_TOTALPRICE = SUM(L_EXTENDEDPRICE – L_DISCOUNT) * (1 + L_TAX)*
For each ORDER and LINEITEM defined by (O_ORDERKEY = L_ORDERKEY)

The following queries were executed before and after a measurement to show that the database was always in a consistent state both initially and after a measurement.

SELECT DECIMAL (SUM (DECIMAL (INTEGER (INTEGER (DECIMAL (INTEGER (100 * DECIMAL (L_EXTENDEDPRICE, 20, 3)), 20, 3) * (1 – L_DISCOUNT)) * (1 + L_TAX)), 20, 3) / 100.0) 20, 3) FROM TPCD.LINEITEM WHERE L_ORDERKEY = okey

© 2004 Hewlett Packard Company. All rights reserved

SELECT DECIMAL(SUM(O_TOTALPRICE, 20, 3)) from TPCH.ORDERS WHERE O_ORDERKEY = okey

4.2.1 Consistency Tests

Verify that ORDER and LINEITEM tables are initially consistent as defined in Clause 3.3.2.1, based upon a random sample of at least 10 distinct values of O_ORDERKEY.

The following steps were performed to verify the Consistency of ACID transactions:

1. The consistency of the ORDERS and LINEITEM tables was verified based on a sample of order keys.
2. 100 ACID Transactions were submitted from each of 8 execution streams.
3. The consistency of the ORDERS and LINEITEM tables was re-verified.

4.3 Isolation Requirements

Operations of concurrent transactions must yield results which are indistinguishable from the results which would be obtained by forcing each transaction to be serially executed to completion in some order.

4.3.1 Isolation Test 1 - Read-Write Conflict with Commit

Demonstrate isolation for the read-write conflict of a read-write transaction and a read-only transaction when the read-write transaction is committed.

The following steps were performed to satisfy the test of isolation for a read-only and a read-write committed transaction:

1. An ACID Transaction was started for a randomly selected O_KEY, L_KEY, and DELTA. The ACID Transaction was suspended prior to COMMIT.
2. An ACID Query was started for the same O_KEY used in step 1. The ACID Query blocked and did not see any uncommitted changes made by the ACID Transaction.
3. The ACID Transaction was resumed, and COMMITTED.
4. The ACID Query completed. It returned the data as committed by the ACID Transaction.

4.3.2 Isolation Test 2 - Read-Write Conflict with Rollback

Demonstrate isolation for the read-write conflict of a read-write transaction and a read-only transaction when the read-write transaction is rolled back.

The following steps were performed to satisfy the test of isolation for a read-only and a rolled back read-write transaction:

1. An ACID Transaction was started for a randomly selected O_KEY, L_KEY, and DELTA. The ACID Transaction was suspended prior to ROLLBACK.
2. An ACID Query was started for the same O_KEY used in step 1. The ACID Query did not see the uncommitted changes made by the ACID Transaction.
3. The ACID Transaction was ROLLED BACK.
4. The ACID Query completed.

4.3.3 Isolation Test 3 - Write-Write Conflict with Commit

Demonstrate isolation for the write-write conflict of two update transactions when the first transaction is committed.

The following steps were performed to verify isolation of two update transactions:

1. An ACID Transaction, T1, was started for a randomly selected O_KEY, L_KEY, and DELTA. The ACID transaction T1 was suspended prior to COMMIT.

© 2004 Hewlett Packard Company. All rights reserved

2. Another ACID Transaction, T2, was started using the same O_KEY and L_KEY and a randomly selected DELTA.
3. T2 waited.
4. T1 was allowed to COMMIT and T2 completed.
5. It was verified that T2.L_EXTENDEDPRICE = T1.L_EXTENDEDPRICE + (DELTA1*(T1.L_EXTENDEDPRICE/T1.L_QUANTITY))

4.3.4 Isolation Test 4 - Write-Write Conflict with Rollback

Demonstrate isolation for the write-write conflict of two update transactions when the first transaction is rolled back.

The following steps were performed to verify isolation of two update transactions after the first one is rolled back:

1. An ACID Transaction, T1, was started for a randomly selected O_KEY, L_KEY, and DELTA. The ACID transaction T1 was suspended prior to ROLLBACK.
2. Another ACID Transaction, T2, was started using the same O_KEY and L_KEY and a randomly selected DELTA.
3. T2 waited.
4. T1 was allowed to ROLLBACK and T2 completed.
5. It was verified that T2.L_EXTENDEDPRICE = T1.L_EXTENDEDPRICE.

4.3.5 Isolation Test 5 – Concurrent Read and Write Transactions on Different Tables

Demonstrate the ability of read and write transactions affecting different database tables to make progress concurrently.

The following steps were performed to demonstrate the ability of read and write transactions affecting different database tables to make progress concurrently:

1. An ACID Transaction, T1, was started for a randomly selected O_KEY, L_KEY, and DELTA. T1 was suspended prior to COMMIT.
2. Another ACID transaction, T2 was started using random values for PS_PARTKEY and PS_SUPPKEY, all columns of the PARTSUPP table for which PS_PARTKEY and PS_SUPPKEy are equal are returned.
3. ACID Transaction T2 completed.
4. T1 was allowed to COMMIT.
5. It was verified that the appropriate rows in the ORDER, LINEITEM, and HISTORY tables have been changed.

4.3.6 Isolation Test 6 – Update Transactions during Continuous Read-Only Query Stream

Demonstrate the continuous submission of arbitrary (read-only) queries against one or more tables of the database does not indefinitely delay update transactions affecting those tables from making progress.

The following steps were performed to demonstrate that the continuous submission of arbitrary (read-only) queries against one or more tables of the database:

1. A Transaction, T1, was started which executed Q21 against the qualification database, was started using a randomly selected DELTA.
2. An ACID Transaction, T2, was started for a randomly selected O_KEY, L_KEY and DELTA.
3. T2 completed and appropriate rows in the ORDERS, LINEITEM and HISTORY tables had been changed.
4. Transaction T1 completed executing Q21.

4.4 Durability Requirements

The tested system must guarantee durability: the ability to preserve the effects of committed transactions and insure database consistency after recovery from any one of the failures listed in Clause 3.5.2.

© 2004 Hewlett Packard Company. All rights reserved

4.4.1 Permanent Unrecoverable Failure of Any Durable Medium

Guarantee the database and committed updates are preserved across a permanent irrecoverable failure of any single durable medium containing TPC-H database tables or recovery log tables.

During the test one disk from one of redo log volume was removed; as the redo log files resided on RAID1+0 volumes test continued uninterrupted

One of the data files was backup up. During the test this data file was corrupted by copying garbage information onto it. The database reported file corruption and instances were terminated. The data file was restored from the backup. Started Oracle instance, which automatically recovered the database. Consistency conditions were verified.

4.4.2 System Crash

Guarantee the database and committed updates are preserved across an instantaneous interruption (system crash/system hang) in processing which requires the system to reboot to recover.

The system crash and memory failure tests were combined. Power to the servers was turned off during the test. The power was restored and database was restarted. The durability success file and the HISTORY table were compared and the counts matched.

4.4.3 Memory Failure

Guarantee the database and committed updates are preserved across failure of all or part of memory (loss of contents).

The system crash and memory failure tests were combined. Power to the servers was turned off during the test. The power was restored and database was restarted. The durability success file and the HISTORY table were compared and the counts matched.

© 2004 Hewlett Packard Company. All rights reserved

5.0 Clause 4: Scaling and Database Population

5.1 Initial Cardinality of Tables

The cardinality (e.g., the number of rows) of each table of the test database, as it existed at the completion of the database load (see clause 4.2.5) must be disclosed.

Table 5.1 lists the TPC Benchmark H defined tables and the row count for each table as they existed upon completion of the build.

Table 5. 1: Initial Number of Rows

Table Name	Row Count
Region	5
Nation	25
Supplier	10,000,000
Customer	150,000,000
Part	200,000,000
Partsupp	800,000,000
Orders	1,500,000,000
Lineitem	5,999,994,267

5.2 Distribution of Tables and Logs Across Media

The distribution of tables and logs across all media must be explicitly described for the tested and priced systems.

The HP ProLiant DL585 Cluster 48P is depicted in Figure 1.1 consisted of:

Processors (per node) :	4 x 2.2GHz/1MB AMD Opteron Model 848
Memory (per node) :	8 GB
OS Disk Drives (per node) :	2 x 36GB 15krpm HDD Ultra320
NICs (per node) :	2 x on-board
Cluster interconnect (per node) :	1 x InfiniServ 7000 (connected to InfinIO 3016 switch)
Disk Controllers (per node) :	6 x hp StorageWorks fca 2214DC
Storage Area Network :	12 x hp StorageWorks SAN Switch 2/16
	48 x hp StorageWorks MSA1000
	408 x 36GB 15krpm HDD Ultra320
Total Storage :	14688GB

The storage area network (SAN) consisted of 12 hp SAN Switch 2/16s and 48 hp StorageWorks MSA1000s. The hp ProLiant DL585 servers had six hp StorageWorks fca 2214DC (dual port) HBAs; each port connected to one of twelve hp SAN Switch 2/16s. Each hp SAN Switch 2/16s had four hp StorageWorks MSA1000s connected to it.

Each hp StorageWorks MSA1000 had two RAID0 volumes of four 36GB 15krpm HDD Ultra320 each for data; 12 MSA 1000s had one additional RAID1+0 volume of two 36GB 15krpm HDD Ultra320 each

© 2004 Hewlett Packard Company. All rights reserved

for redo logs. The array accelerator cache was set to 80% read 20% write for data volumes and disabled for redo log volumes.

All tables except NATION and REGION had 96 horizontal partitions. The temp files and flat files were also distributed across the 96 volumes, two per hp StorageWorks MSA1000s.

One InfiniCon InfinIO 3016 switch (Infiniband) switch was used as cluster interconnect.

A detailed description of distribution of database files can be found in Table 5.2.

Table 5.2: SAN configuration and Database Layout

© 2004 Hewlett Packard Company. All rights reserved

SAN Switch	SAN Array	Logical Unit	Partition 1	Partitions 2	Partition 3	Partition 5
1	1	1	redo log	redo log	ocr	quorum
1	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
1	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
1	2	1	lineitem, order, customer, part, supplier, partsupp	temp	sys	flat files
1	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
1	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
1	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
1	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
1	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
2	1	1	redo log	redo log		
2	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
2	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
2	2	1	lineitem, order, customer, part, supplier, partsupp	temp	sys	flat files
2	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
2	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
2	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
2	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
2	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
3	1	1	redo log	redo log		
3	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
3	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
3	2	1	lineitem, order, customer, part, supplier, partsupp	temp	perf	flat files
3	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
3	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
3	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
3	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
3	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
4	1	1	redo log			
4	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
4	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
4	2	1	lineitem, order, customer, part, supplier, partsupp	temp	sysaux	flat files
4	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
4	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
4	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
4	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
4	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
5	1	1	redo log	redo log		
5	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
5	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
5	2	1	lineitem, order, customer, part, supplier, partsupp	temp	default	flat files
5	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
5	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
5	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
5	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
5	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
6	1	1	redo log	redo log		
6	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
6	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
6	2	1	lineitem, order, customer, part, supplier, partsupp	temp	control	flat files
6	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
6	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
6	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
6	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
6	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
7	1	1	redo log	redo log		
7	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
7	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
7	2	1	lineitem, order, customer, part, supplier, partsupp	temp	control	flat files
7	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
7	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
7	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
7	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
7	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
8	1	1	redo log	redo log		
8	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
8	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
8	2	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
8	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
8	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
8	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
8	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
8	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
9	1	1	redo log	redo log		
9	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
9	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
9	2	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
9	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
9	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
9	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
9	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
9	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
10	1	1	redo log	redo log		
10	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
10	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
10	2	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
10	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
10	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
10	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
10	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
10	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
11	1	1	redo log			
11	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
11	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
11	2	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
11	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
11	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
11	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
11	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
11	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
12	1	1	redo log	redo log		
12	1	2	lineitem, order, customer, part, supplier, partsupp	temp	undo	flat files
12	1	3	lineitem, order, customer, part, supplier, partsupp	temp	1gb - qualification db	flat files
12	2	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
12	2	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
12	3	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
12	3	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files
12	4	1	lineitem, order, customer, part, supplier, partsupp	temp		flat files
12	4	2	lineitem, order, customer, part, supplier, partsupp	temp		flat files

5.3 Mapping of Database Partitions/Replications

The mapping of database partitions/replications must be explicitly described.

© 2004 Hewlett Packard Company. All rights reserved

Horizontal partitioning was used for all tables except NATION and REGION. Sections 5.2 describe the distribution of tables and redo log files. The database was not replicated.

5.4 Implementation of RAID

Implementations may use some form of RAID to ensure high availability. If used for data, auxiliary storage (e.g. indexes) or temporary space, the level of RAID used must be disclosed for each device.

RAID 0+1 was used for the redo log files, system and auxiliary tablespaces, and database control files. RAID0 (striping) was used for the rest of the database datafiles.

5.5 DBGEN Modifications

The version number, release number, modification number, and patch level of DBGEN must be disclosed. Any modifications to the DBGEN (see Clause 4.2.1) source code must be disclosed. In the event that a program other than DBGEN was used to populate the database, it must be disclosed in its entirety.

The supplied DBGEN version 1.3.0 was used to generate the database population for this benchmark without any modification.

5.6 Database Load time

The database load time for the test database (see clause 4.3) must be disclosed.

The database load time was 4 hours 12 minutes 23 seconds including the elapse time for the backup.

5.7 Data Storage Ratio

The data storage ratio must be disclosed. It is computed by dividing the total data storage of the priced configuration (expressed in GB) by the size chosen for the test database as defined in 4.1.3.1. The ratio must be reported to the nearest 1/100th, rounded up.

Disk Type	Number of Disks	Total Disk Space	Data Storage Ratio
36GB15krpm HDD Ultra320	408	14688GB	14.688

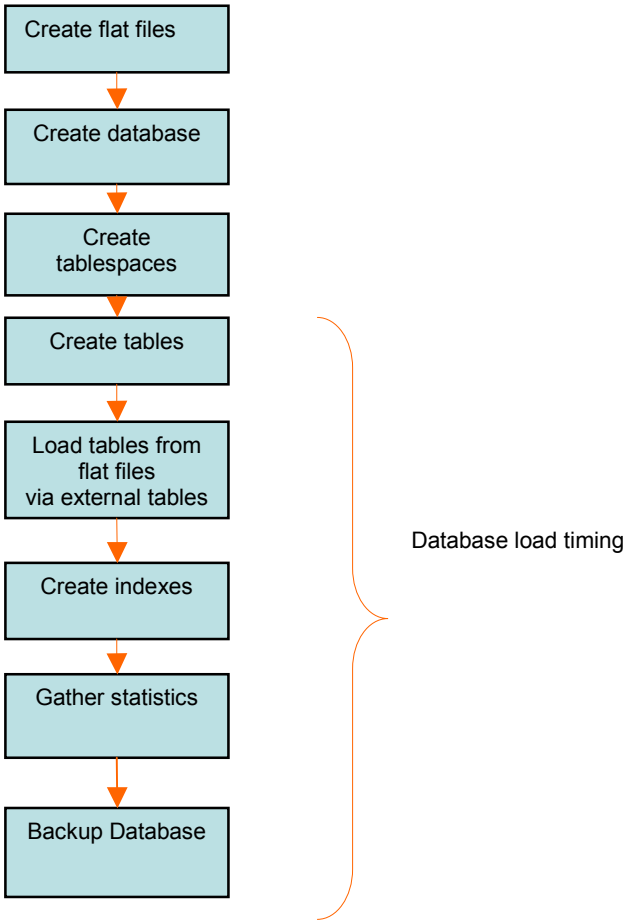
© 2004 Hewlett Packard Company. All rights reserved

5.8 Database Load Mechanism Details and Illustration

The details of the database load must be disclosed, including a block diagram illustrating the overall process. Disclosure of the load procedure includes all steps, scripts, input and configuration files required to completely reproduce the test and qualification databases.

Flat files for each of the tables were created using DBGEN resided on the SAN.

Figure 5.8: Block Diagram of Database Load Process



5.9 Qualification Database Configuration

Any differences between the configuration of the qualification database and the test database must be disclosed.

The qualification database used identical scripts to create and load the data with changes to adjust for the database scale factor.

© 2004 Hewlett Packard Company. All rights reserved

6.0 Clause 5: Performance Metrics and Execution Rules Related Items

6.1 Steps in the Power Test

The details of the steps followed to implement the power test (e.g., system boot, database restart, etc.) must be disclosed.

The following steps were used to implement the power test:

1. The system was rebooted
2. RF1 Refresh Transaction
3. Stream 00 Execution
4. RF2 Refresh Transaction.

6.2 Timing Intervals for Each Query and Refresh Function

The timing intervals (see Clause 5.3.6) for each query of the measured set and for both refresh functions must be reported for the power test.

Numerical Quantities Summary section of the executive summary, which can be found in the beginning of this document, contains the timing intervals for queries and refresh functions.

6.3 Number of Streams for The Throughput Test

The number of execution streams used for the throughput test must be disclosed.

Eight streams were used for the Throughput Test.

6.4 Start and End Date/Times for Each Query Stream

The start time and finish time for each query execution stream must be reported for the throughput test.

Numerical Quantities Summary section of the executive summary, which can be found in the beginning of this document, contains the start and stop times for the query execution streams.

6.5 Total Elapsed Time for the Measurement Interval

The total elapsed time of the measurement interval (see Clause 5.3.5) must be reported for the throughput test.

Numerical Quantities Summary section of the executive summary, which can be found in the beginning of this document, contains the elapsed time for the measurement interval.

6.6 Refresh Function Start Date/Time and Finish Date/Time

Start and finish time for each update function in the update stream must be reported for the throughput test.

Numerical Quantities Summary section of the executive summary, which can be found in the beginning of this document, contains the start and finish time for the refresh functions.

6.7 Timing Intervals for Each Query and Each Refresh Function for Each Stream

The timing intervals (see Clause 5.3.6) for each query of each stream and for each update function must be reported for the throughput test.

© 2004 Hewlett Packard Company. All rights reserved

Numerical Quantities Summary section of the executive summary, which can be found in the beginning of this document, contains the timing intervals for queries and refresh functions.

6.8 Performance Metrics

The computed performance metrics, related numerical quantities and the price performance metric must be reported.

Numerical Quantities Summary section of the executive summary, which can be found in the beginning of this document, contains the performance metrics, related numerical quantities and the price performance metric.

6.9 The Performance Metric and Numerical Quantities from Both Runs

A description of the method used to determine the reproducibility of the measurement results must be reported. This must include the performance metrics (QppH and QthH) from the reproducibility runs.

Performance results from the first two executions of the TPC-H benchmark indicated the following difference for the metric points:

Run	QppH@1TB	QthH@1TB	QphH@1TB
Run 1	46463.6	26577.6	35141.0
Run 2	47145.1	26794.9	35542.2

6.11 System Activity Between Tests

Any activity on the SUT that takes place between the conclusion of Run1 and the beginning of Run2 must be disclosed.

The database was restarted between the two runs.
The system was restarted between runs.

© 2004 Hewlett Packard Company. All rights reserved

7.0 Clause 6: SUT and Driver Implementation Related Items

7.1 Driver

A detailed description of how the driver performs its functions must be supplied, including any related source code or scripts. This description should allow an independent reconstruction of the driver.

A single script performs all stream executions. QGEN is used to produce query text. For each power-test run:
The SQL for RF1 is submitted to the database
Then the queries as generated by QGEN are submitted in the order defined by Clause 5.3.5.4
The SQL for RF2 is submitted to the database.

7.2 Implementation Specific Layer (ISL)

If an implementation-specific layer is used, then a detailed description of how it performs its functions must be supplied, including any related source code or scripts. This description should allow an independent reconstruction of the implementation-specific layer.

The source code for the qexec utility can be found in Appendix F.

7.3 Profile-Directed Optimization

If profile-directed optimization as described in Clause 5.2.9 is used, such used must be disclosed.

Profile-directed optimization was used in this benchmark.

© 2004 Hewlett Packard Company. All rights reserved

8.0 Clause 7: Pricing Related Items

8.1 Hardware and Software Used

A detailed list of hardware and software used in the priced system must be reported. Each item must have vendor part number, description, and release/revision level, and either general availability status or committed delivery date. If package-pricing is used, contents of the package must be disclosed. Pricing source(s) and effective date(s) of price(s) must also be reported.

A detailed list of hardware and software used in the priced system is included in the pricing sheet in the executive summary. All prices are currently effective. Third-party price quotations are included in Appendix G.

8.2 Total 3 Year Price

The total 3-year price of the entire configuration must be reported including: hardware, software, and maintenance charges. Separate component pricing is recommended. The basis of all discounts used must be disclosed.

A detailed pricing sheet of all the hardware and software used in this configuration and the 3-year maintenance costs, demonstrating the computation of the total 3-year price of the configuration, is included in the executive summary at the beginning of this document. For a large purchase and cash discount, this purchase qualifies for a 17% discount from Hewlett Packard Company.

8.3 Availability Date

The committed delivery date for general availability of products used in the price calculations must be reported. When the priced system includes products with different availability dates, the availability date reported on the executive summary must be the date by which all components are committed to being available. The full disclosure report must report availability dates individually for at least each of the categories for which a pricing subtotal must be provided.

All hardware and software are available at the time of publication.

8.4 Country-Specific Pricing

Additional Clause 7 related items may be included in the Full Disclosure Report for each country-specific priced configuration. Country-specific pricing is subject to Clause 7.1.7.

The configuration is priced for the United States of America.

© 2004 Hewlett Packard Company. All rights reserved

9.0 Clause 9: Related Items

9.1 Auditors' Report

The auditor's agency name, address, phone number, and Attestation letter with a brief audit summary report indicating compliance must be included in the full disclosure report. A statement should be included specifying who to contact in order to obtain further information regarding the audit process.

Lorna Livingtree of Performance Metrics Inc audited this implementation of the TPC Benchmark H.

Performance Metrics Inc.
137 Yankton St., Suite 101
Folsom, CA 95630
USA

Email: lorna@perfmetrics.com
Phone: (707) 482-0523

TPC Benchmark H Full Disclosure Report and other information can be downloaded from Transaction Processing Performance Council web site at www.tpc.org.

© 2004 Hewlett Packard Company. All rights reserved

Appendix A: Tunable Parameters

```
_____________________________________
init_h10.ora
_____________________________________
instance_number =10
thread  =10
undo_management              = auto
UNDO_TABLESPACE              = ts_undo10
cluster_database             = true
#cluster_interconnects       = 10.2.0.10
cluster_interconnects        = 10.3.0.10
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h11.ora
_____________________________________
instance_number =11
thread  =11
undo_management              = auto
UNDO_TABLESPACE              = ts_undo11
cluster_database             = true
#cluster_interconnects       = 10.2.0.11
cluster_interconnects        = 10.3.0.11
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h12.ora
_____________________________________
instance_number =12
thread  =12
undo_management              = auto
UNDO_TABLESPACE              = ts_undo12
cluster_database             = true
#cluster_interconnects       = 10.2.0.12
cluster_interconnects        = 10.3.0.12
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h1.ora
_____________________________________
instance_number =1
thread  =1
undo_management              = auto
UNDO_TABLESPACE              = ts_undo1
cluster_database             = true
#10.2.0.1 = GIGE
#10.3.0.1 = IB
#cluster_interconnects       = 10.2.0.1
cluster_interconnects        = 10.3.0.1
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h2.ora
_____________________________________
instance_number =2
thread  =2
undo_management              = auto
UNDO_TABLESPACE              = ts_undo2
cluster_database             = true
#cluster_interconnects       = 10.2.0.2
cluster_interconnects        = 10.3.0.2
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h3.ora
_____________________________________
instance_number =3
thread  =3
undo_management              = auto
UNDO_TABLESPACE              = ts_undo3
cluster_database             = true
#cluster_interconnects       = 10.2.0.3
cluster_interconnects        = 10.3.0.3
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h4.ora
_____________________________________
instance_number =4
thread  =4
undo_management              = auto
UNDO_TABLESPACE              = ts_undo4
cluster_database             = true
#cluster_interconnects       = 10.2.0.4
cluster_interconnects        = 10.3.0.4
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h5.ora
_____________________________________
instance_number =5
thread  =5
undo_management              = auto
UNDO_TABLESPACE              = ts_undo5
cluster_database             = true
#cluster_interconnects       = 10.2.0.5
cluster_interconnects        = 10.3.0.5
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h6.ora
_____________________________________
instance_number =6
thread  =6
undo_management              = auto
UNDO_TABLESPACE              = ts_undo6
cluster_database             = true
#cluster_interconnects       = 10.2.0.6
cluster_interconnects        = 10.3.0.6
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h7.ora
_____________________________________
instance_number =7
thread  =7
undo_management              = auto
UNDO_TABLESPACE              = ts_undo7
cluster_database             = true
#cluster_interconnects       = 10.2.0.7
cluster_interconnects        = 10.3.0.7
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h8.ora
_____________________________________
instance_number =8
thread  =8
undo_management              = auto
UNDO_TABLESPACE              = ts_undo8
cluster_database             = true
#cluster_interconnects       = 10.2.0.8
cluster_interconnects        = 10.3.0.8
ifile=/home/oracle/tpch/dbs/init_run.ora
_____________________________________
init_h9.ora
_____________________________________
instance_number =9
thread  =9
```


© 2004 Hewlett Packard Company. All rights reserved

```
undo_management                  = auto
UNDO_TABLESPACE                  = ts_undo9
cluster_database                 = true
#cluster_interconnects           = 10.2.0.9
cluster_interconnects            = 10.3.0.9
ifile=/home/oracle/tpch/dbs/init_run.ora
```

init_run.ora

```
undo_management                  = auto
aq_tm_processes                  = 0
audit_trail                      = FALSE
compatible                       = 10.0.0.0
control_files                    = (/home/oracle/dev/raw/control_001,
/home/oracle/dev/raw/control_002)
cpu_count                        = 4
db_block_checksum                = false
db_block_size                    = 16384
#db_cache_size                   = 1250m
#db_cache_size                   = 300m
#db_cache_size                   = 1700m
db_cache_size                    = 1500m
db_file_multiblock_read_count    = 128
db_files                         = 500
db_name                          = 10i
db_writer_processes              = 4
dml_locks                        = 40000
enqueue_resources                = 40000
global_names                     = FALSE
instance_name                    = inst_1
#shared_pool_size                = 512m
shared_pool_size                 = 840m  #was 640m
#large_pool_size                 = 300m
log_buffer                       = 4194304
log_checkpoints_to_alert         = true
max_dump_file_size               = unlimited
nls_date_format                  = YY-MM-DD
open_cursors                     = 600
optimizer_mode                   = CHOOSE
optimizer_features_enable        = 10.2.0.1
parallel_adaptive_multi_user     = TRUE
parallel_execution_message_size  = 16384
#parallel_execution_message_size  = 32768
parallel_max_servers             = 140
parallel_min_servers             = 140
pga_aggregate_target             = 6g #6g
processes                        = 1000
query_rewrite_enabled            = true
recovery_parallelism             = 8
replication_dependency_tracking  = false
statistics_level                 = BASIC
transactions                     = 10
undo_retention                   = 500000
```

modules.conf

```
alias eth0 tg3
alias eth1 tg3

# ql2xmaxqdepth=32 ql2xintrdelaytimer=0
#add options scsi_hostadapter ql2xmaxqdepth=32 ql2xintrdelaytimer=0
alias scsi_hostadapter1 cciss
alias usb-controller usb-ohci
# -- All InfiniBand Drivers Start here --
# [ICS VERSION STRING: unknown]
keep
path[iba]=/lib/modules/`uname -r`/iba
#
# InfiniCon Systems (r) InfiniBand(tm) Transport Drivers
# The FixedLID option is used for static LID configuration, when
# its necessary to function without a external Subnet Manager.
# Used only for external loopback testing or HCA to HCA links not
needing
# an SA. (USE WITH CARE !!!)
# options ibt FixedLID=1 BaseLID=6 DefaultLMC=0x0 HopCount=0
# options ibt FixedLID=1 BaseLID=6 DefaultLMC=0x0 HopCount=0
SADisable=1
# Supported Debug Options
#   BusDbg, VcaDbg, SmaDbg, GsaDbg, CmDbg, SdDbg, PmaDbg,
IbtDbg
#   SqptDbg
#      the above enable debug output from different modules of the
#      transport driver.
#   Log2Cons
#   PmaDisable, SADisable
#    cat /proc/iba/config for more information
# options ibt SADisable=1
options ibt
#options ibt CmMaxDgrms=15000 CmReuseQP=1
GsiMinRecvPerPort=1000 GsiPreallocRecvPerPort=1200
CmMinTurnaroundTime=1000
# Verbs Provider Driver for Intel(r) (82808XA.o)
# Supported Options
#    Task=1
#    VpdDbg=<flags> to enable Verbs Provider driver debug messages.
options 82808XA Task=1
# Verbs Provider Driver for InfiniCon Systems(r) InfiniServ HCA
(mt23108vpd.o)
# Supported Options:
#    MVpdDbg RnrNakTimer, MaxVLs, MaxMtu, VLArbDisable,
TrapDisable
#    cat /proc/iba/mt23108/config for more information
#options mt23108vpd MaxVLs=0 RatePatch=0 PathbitsPatch=0
SlPatch=0 SmaCache=0 TrapDisable=0 VLArbDisable=0
MKeyCheck=1 TrapLimit=1
# downgrade VLs to 4+1 so that we get a 2K MTU when connected to
Redswitch
#options mt23108vpd MaxVLs=5
options mt23108vpd MaxVLs=5 use_mellanox_locking=1
# Upper Level Protocol Drivers
# InfiniFibre SRP storage driver for Linux
#options ics_srp
below ics_srp scsi_mod sd_mod
# InfiniNIC network driver for Linux
#options ics_inic HostRecvPoolEntries=256
options ics_inic
# AF offload driver for Linux
options ics_offload
# Sdp network driver for Linux
options ics_sdp
# IP over IB network driver for Linux
#options ipoib IpoibXmitBuffers=200 IpoibMCXmitBuffers=200
#options ipoib
#options ipoib IpoibXmitBuffers=100 IpoibRecvBuffers=2000 #2k
bytes
options ipoib IpoibXmitBuffers=100 IpoibRecvBuffers=2000
# Discovery support driver for Linux
# Disable Trap Notification Report Subscription with the SM
# therefore Scan Fabric every 10 seconds
#options ics_dsc dsc_no_subscribe=1 dsc_scan_frequency=10
options ics_dsc
# modules depending on 82808XA
pre-remove 82808XA modprobe -r ics_srp && modprobe -r ics_inic
&& modprobe -r ics_sdp && modprobe -r ics_offload && modprobe -r
ipoib && modprobe -r ics_dsc
# modules depending on mt23108vpd
```


© 2004 Hewlett Packard Company. All rights reserved

```
pre-remove mt23108vpd modprobe -r ics_srp && modprobe -r ics_inic
&& modprobe -r ics_sdp && modprobe -r ics_offload && modprobe -r
ipoib && modprobe -r ics_dsc
# -- All InfiniBand Drivers End here --
#post-remove qla2300 rmmod qla2300_conf
#alias scsi_hostadapter2 qla2300_conf
alias scsi_hostadapter3 qla2300
#alias scsi_hostadapter4 sg
options qla2300  ql2xmaxqdepth=24 ql2xintrdelaytimer=0
#options qla2300  #add ql2xintrdelaytimer=0 ql2xmaxqdepth=16
qlport_down_retry=64 qlogin_retry_count=16 ql2xfailover=0
alias scsi_hostadapter qla2300
```

rc.local

```
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

touch /var/lock/subsys/local

#rmmod qla2300
# defaults queue=64 timer=3
# Andy suggests q=32 t=0
#insmod qla2300 ql2xmaxqdepth=32 ql2xintrdelaytimer=0
#insmod qla2300 ql2xintrdelaytimer=0

sh /root/mkraw.sh

rdate -s dbgate

#echo  6442450944 > /proc/sys/kernel/shmmax
echo  3000000000 > /proc/sys/kernel/shmmax
echo  1048576 > /proc/sys/fs/aio-max-nr


#/etc/init.d/netdump start
#ifdown eth0
#ifdown eth1
#ifdown eth2
#rmmod tg3
#insmod /root/bcm5700.o
#ifup eth0
#ifup eth2
#insmod /lib/modules/2.4.21-
1.1931.2.421.entcustom/kernel/drivers/net/e1000/e1000.o

#/sbin/ifconfig eth0 txqueuelen 10000
#/sbin/ifconfig eth1 txqueuelen 10000
#/sbin/ifconfig eth2 txqueuelen 10000


#echo 10 > /proc/irq/20/smp_affinity
#echo 20 > /proc/irq/21/smp_affinity
#echo 40 > /proc/irq/22/smp_affinity
#echo 80 > /proc/irq/23/smp_affinity
#echo 10 > /proc/irq/27/smp_affinity
#echo 20 > /proc/irq/29/smp_affinity
#echo 40 > /proc/irq/30/smp_affinity
#echo 80 > /proc/irq/31/smp_affinity
#echo 10 > /proc/irq/44/smp_affinity
#echo 20 > /proc/irq/43/smp_affinity
#echo 40 > /proc/irq/25/smp_affinity
#

#ifconfig eth2 mtu 9000



sh /root/mkraw.sh
sh /root/mnt.sh
```

sysctl.conf

```
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Controls the System Request debugging functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the PID to the core
filename.
# Useful for debugging multi-threaded applications.
kernel.core_uses_pid = 1

kernel.sem = 250        32000   100      128

net.core.rmem_max = 5242880
net.core.rmem_default = 5242880
net.core.wmem_max = 5242880
net.core.wmem_default = 5242880
net.ipv4.ip_local_port_range = 32768    65000

#net.core.rmem_max = 1048576
#net.core.rmem_default = 1048576
#net.core.wmem_max = 1048576
#net.core.wmem_default = 1048576
#net.ipv4.ip_local_port_range = 32768    65000

fs.aio-max-size = 2097152
kernel.shmmax = 3000000000
#kernel.shmmax = 2147483648 ##default on AS3.0 update 1
```


© 2004 Hewlett Packard Company. All rights reserved

Appendix B: Database Build Scripts

anlyz.sh

```
#!/bin/ksh
echo Analyze Phase
sqlplus /NOLOG <<!
connect tpch/tpch;
!date
set echo on
set timing on
execute dbms_stats.gather_schema_stats('TPCH' , estimate_percent =>
1, degree => 96 , granularity => 'GLOBAL' );
connect / as sysdba
execute dbms_stats.gather_system_stats;
exec dbms_scheduler.disable('GATHER_STATS_JOB');
alter system switch logfile;
!date
exit
!
```

dapop.sh

```
#!/bin/ksh
sqlplus /NOLOG << !
connect / as sysdba
drop user tpch cascade;
grant DBA
to tpch identified by tpch;
connect tpch/tpch;
drop directory data_dir;
create directory data_dir as '/home/oracle/OraHome1/dbs/flatfiles';
create directory d1 as '/home/oracle/dev/ff_1';
create directory d2 as '/home/oracle/dev/ff_2';
create directory d3 as '/home/oracle/dev/ff_3';
create directory d4 as '/home/oracle/dev/ff_4';
create directory d5 as '/home/oracle/dev/ff_5';
create directory d6 as '/home/oracle/dev/ff_6';
create directory d7 as '/home/oracle/dev/ff_7';
create directory d8 as '/home/oracle/dev/ff_8';
create directory d9 as '/home/oracle/dev/ff_9';
create directory d10 as '/home/oracle/dev/ff_10';
create directory d11 as '/home/oracle/dev/ff_11';
create directory d12 as '/home/oracle/dev/ff_12';
create directory d13 as '/home/oracle/dev/ff_13';
create directory d14 as '/home/oracle/dev/ff_14';
create directory d15 as '/home/oracle/dev/ff_15';
create directory d16 as '/home/oracle/dev/ff_16';
create directory d17 as '/home/oracle/dev/ff_17';
create directory d18 as '/home/oracle/dev/ff_18';
create directory d19 as '/home/oracle/dev/ff_19';
create directory d20 as '/home/oracle/dev/ff_20';
create directory d21 as '/home/oracle/dev/ff_21';
create directory d22 as '/home/oracle/dev/ff_22';
create directory d23 as '/home/oracle/dev/ff_23';
create directory d24 as '/home/oracle/dev/ff_24';
create directory d25 as '/home/oracle/dev/ff_25';
create directory d26 as '/home/oracle/dev/ff_26';
create directory d27 as '/home/oracle/dev/ff_27';
create directory d28 as '/home/oracle/dev/ff_28';
create directory d29 as '/home/oracle/dev/ff_29';
create directory d30 as '/home/oracle/dev/ff_30';
create directory d31 as '/home/oracle/dev/ff_31';
create directory d32 as '/home/oracle/dev/ff_32';
create directory d33 as '/home/oracle/dev/ff_33';
create directory d34 as '/home/oracle/dev/ff_34';
create directory d35 as '/home/oracle/dev/ff_35';
create directory d36 as '/home/oracle/dev/ff_36';
create directory d37 as '/home/oracle/dev/ff_37';
create directory d38 as '/home/oracle/dev/ff_38';
create directory d39 as '/home/oracle/dev/ff_39';
create directory d40 as '/home/oracle/dev/ff_40';
create directory d41 as '/home/oracle/dev/ff_41';
create directory d42 as '/home/oracle/dev/ff_42';
create directory d43 as '/home/oracle/dev/ff_43';
create directory d44 as '/home/oracle/dev/ff_44';
create directory d45 as '/home/oracle/dev/ff_45';
create directory d46 as '/home/oracle/dev/ff_46';
create directory d47 as '/home/oracle/dev/ff_47';
create directory d48 as '/home/oracle/dev/ff_48';
create directory d49 as '/home/oracle/dev/ff_49';
create directory d50 as '/home/oracle/dev/ff_50';
create directory d51 as '/home/oracle/dev/ff_51';
create directory d52 as '/home/oracle/dev/ff_52';
create directory d53 as '/home/oracle/dev/ff_53';
create directory d54 as '/home/oracle/dev/ff_54';
create directory d55 as '/home/oracle/dev/ff_55';
create directory d56 as '/home/oracle/dev/ff_56';
create directory d57 as '/home/oracle/dev/ff_57';
create directory d58 as '/home/oracle/dev/ff_58';
create directory d59 as '/home/oracle/dev/ff_59';
create directory d60 as '/home/oracle/dev/ff_60';
create directory d61 as '/home/oracle/dev/ff_61';
create directory d62 as '/home/oracle/dev/ff_62';
create directory d63 as '/home/oracle/dev/ff_63';
create directory d64 as '/home/oracle/dev/ff_64';
create directory d65 as '/home/oracle/dev/ff_65';
create directory d66 as '/home/oracle/dev/ff_66';
create directory d67 as '/home/oracle/dev/ff_67';
create directory d68 as '/home/oracle/dev/ff_68';
create directory d69 as '/home/oracle/dev/ff_69';
create directory d70 as '/home/oracle/dev/ff_70';
create directory d71 as '/home/oracle/dev/ff_71';
create directory d72 as '/home/oracle/dev/ff_72';
create directory d73 as '/home/oracle/dev/ff_73';
create directory d74 as '/home/oracle/dev/ff_74';
create directory d75 as '/home/oracle/dev/ff_75';
create directory d76 as '/home/oracle/dev/ff_76';
create directory d77 as '/home/oracle/dev/ff_77';
create directory d78 as '/home/oracle/dev/ff_78';
create directory d79 as '/home/oracle/dev/ff_79';
create directory d80 as '/home/oracle/dev/ff_80';
create directory d81 as '/home/oracle/dev/ff_81';
create directory d82 as '/home/oracle/dev/ff_82';
create directory d83 as '/home/oracle/dev/ff_83';
create directory d84 as '/home/oracle/dev/ff_84';
create directory d85 as '/home/oracle/dev/ff_85';
create directory d86 as '/home/oracle/dev/ff_86';
create directory d87 as '/home/oracle/dev/ff_87';
create directory d88 as '/home/oracle/dev/ff_88';
create directory d89 as '/home/oracle/dev/ff_89';
create directory d90 as '/home/oracle/dev/ff_90';
create directory d91 as '/home/oracle/dev/ff_91';
create directory d92 as '/home/oracle/dev/ff_92';
create directory d93 as '/home/oracle/dev/ff_93';
create directory d94 as '/home/oracle/dev/ff_94';
create directory d95 as '/home/oracle/dev/ff_95';
create directory d96 as '/home/oracle/dev/ff_96';
```


© 2004 Hewlett Packard Company. All rights reserved

```
drop table l_et;
create table l_et(
   l_orderkey          number ,
   l_partkey           number ,
   l_suppkey           number ,
   l_linenumber        number ,
   l_quantity          number ,
   l_extendedprice     number ,
   l_discount          number ,
   l_tax               number ,
   l_returnflag        char(1) ,
   l_linestatus        char(1) ,
   l_shipdate          date ,
   l_commitdate        date ,
   l_receiptdate       date ,
   l_shipinstruct      char(25) ,
   l_shipmode          char(10) ,
   l_comment           varchar(44)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
                        records delimited by newline
                        nobadfile
                        nologfile
                        fields terminated by '|'
                        missing field values are null
            )
            location (
d1:'lineitem.tbl.1',d2:'lineitem.tbl.2',
d3:'lineitem.tbl.3',d4:'lineitem.tbl.4',
d5:'lineitem.tbl.5',d6:'lineitem.tbl.6',
d7:'lineitem.tbl.7',d8:'lineitem.tbl.8',
d9:'lineitem.tbl.9',d10:'lineitem.tbl.10',
d11:'lineitem.tbl.11',d12:'lineitem.tbl.12',
d13:'lineitem.tbl.13',d14:'lineitem.tbl.14',
d15:'lineitem.tbl.15',d16:'lineitem.tbl.16',
d17:'lineitem.tbl.17',d18:'lineitem.tbl.18',
d19:'lineitem.tbl.19',d20:'lineitem.tbl.20',
d21:'lineitem.tbl.21',d22:'lineitem.tbl.22',
d23:'lineitem.tbl.23',d24:'lineitem.tbl.24',
d25:'lineitem.tbl.25',d26:'lineitem.tbl.26',
d27:'lineitem.tbl.27',d28:'lineitem.tbl.28',
d29:'lineitem.tbl.29',d30:'lineitem.tbl.30',
d31:'lineitem.tbl.31',d32:'lineitem.tbl.32',
d33:'lineitem.tbl.33',d34:'lineitem.tbl.34',
d35:'lineitem.tbl.35',d36:'lineitem.tbl.36',
d37:'lineitem.tbl.37',d38:'lineitem.tbl.38',
d39:'lineitem.tbl.39',d40:'lineitem.tbl.40',
d41:'lineitem.tbl.41',d42:'lineitem.tbl.42',
d43:'lineitem.tbl.43',d44:'lineitem.tbl.44',
d45:'lineitem.tbl.45',d46:'lineitem.tbl.46',
d47:'lineitem.tbl.47',d48:'lineitem.tbl.48',
d49:'lineitem.tbl.49',d50:'lineitem.tbl.50',
d51:'lineitem.tbl.51',d52:'lineitem.tbl.52',
d53:'lineitem.tbl.53',d54:'lineitem.tbl.54',
d55:'lineitem.tbl.55',d56:'lineitem.tbl.56',
d57:'lineitem.tbl.57',d58:'lineitem.tbl.58',
d59:'lineitem.tbl.59',d60:'lineitem.tbl.60',
d61:'lineitem.tbl.61',d62:'lineitem.tbl.62',
d63:'lineitem.tbl.63',d64:'lineitem.tbl.64',
d65:'lineitem.tbl.65',d66:'lineitem.tbl.66',
d67:'lineitem.tbl.67',d68:'lineitem.tbl.68',
d69:'lineitem.tbl.69',d70:'lineitem.tbl.70',
d71:'lineitem.tbl.71',d72:'lineitem.tbl.72',
d73:'lineitem.tbl.73',d74:'lineitem.tbl.74',
d75:'lineitem.tbl.75',d76:'lineitem.tbl.76',
d77:'lineitem.tbl.77',d78:'lineitem.tbl.78',
d79:'lineitem.tbl.79',d80:'lineitem.tbl.80',
d81:'lineitem.tbl.81',d82:'lineitem.tbl.82',
d83:'lineitem.tbl.83',d84:'lineitem.tbl.84',
d85:'lineitem.tbl.85',d86:'lineitem.tbl.86',
d87:'lineitem.tbl.87',d88:'lineitem.tbl.88',
d89:'lineitem.tbl.89',d90:'lineitem.tbl.90',
d91:'lineitem.tbl.91',d92:'lineitem.tbl.92',
d93:'lineitem.tbl.93',d94:'lineitem.tbl.94',
d95:'lineitem.tbl.95',d96:'lineitem.tbl.96'
))
reject limit unlimited;
drop table o_et;
create table o_et(
   o_orderkey          number ,
   o_custkey           number ,
   o_orderstatus       char(1) ,
   o_totalprice        number ,
   o_orderdate         date ,
   o_orderpriority     char(15) ,
   o_clerk             char(15) ,
   o_shippriority      number ,
   o_comment           varchar(79)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
                        records delimited by newline
                        nobadfile
                        nologfile
                        fields terminated by '|'
                        missing field values are null
            )
            location (
d1:'orders.tbl.1',d2:'orders.tbl.2',
d3:'orders.tbl.3',d4:'orders.tbl.4',
d5:'orders.tbl.5',d6:'orders.tbl.6',
d7:'orders.tbl.7',d8:'orders.tbl.8',
d9:'orders.tbl.9',d10:'orders.tbl.10',
d11:'orders.tbl.11',d12:'orders.tbl.12',
d13:'orders.tbl.13',d14:'orders.tbl.14',
d15:'orders.tbl.15',d16:'orders.tbl.16',
d17:'orders.tbl.17',d18:'orders.tbl.18',
d19:'orders.tbl.19',d20:'orders.tbl.20',
d21:'orders.tbl.21',d22:'orders.tbl.22',
d23:'orders.tbl.23',d24:'orders.tbl.24',
d25:'orders.tbl.25',d26:'orders.tbl.26',
d27:'orders.tbl.27',d28:'orders.tbl.28',
d29:'orders.tbl.29',d30:'orders.tbl.30',
d31:'orders.tbl.31',d32:'orders.tbl.32',
d33:'orders.tbl.33',d34:'orders.tbl.34',
d35:'orders.tbl.35',d36:'orders.tbl.36',
d37:'orders.tbl.37',d38:'orders.tbl.38',
d39:'orders.tbl.39',d40:'orders.tbl.40',
d41:'orders.tbl.41',d42:'orders.tbl.42',
d43:'orders.tbl.43',d44:'orders.tbl.44',
d45:'orders.tbl.45',d46:'orders.tbl.46',
d47:'orders.tbl.47',d48:'orders.tbl.48',
d49:'orders.tbl.49',d50:'orders.tbl.50',
d51:'orders.tbl.51',d52:'orders.tbl.52',
d53:'orders.tbl.53',d54:'orders.tbl.54',
d55:'orders.tbl.55',d56:'orders.tbl.56',
d57:'orders.tbl.57',d58:'orders.tbl.58',
d59:'orders.tbl.59',d60:'orders.tbl.60',
d61:'orders.tbl.61',d62:'orders.tbl.62',
d63:'orders.tbl.63',d64:'orders.tbl.64',
d65:'orders.tbl.65',d66:'orders.tbl.66',
d67:'orders.tbl.67',d68:'orders.tbl.68',
d69:'orders.tbl.69',d70:'orders.tbl.70',
```


© 2004 Hewlett Packard Company. All rights reserved

```
d71:'orders.tbl.71',d72:'orders.tbl.72',
d73:'orders.tbl.73',d74:'orders.tbl.74',
d75:'orders.tbl.75',d76:'orders.tbl.76',
d77:'orders.tbl.77',d78:'orders.tbl.78',
d79:'orders.tbl.79',d80:'orders.tbl.80',
d81:'orders.tbl.81',d82:'orders.tbl.82',
d83:'orders.tbl.83',d84:'orders.tbl.84',
d85:'orders.tbl.85',d86:'orders.tbl.86',
d87:'orders.tbl.87',d88:'orders.tbl.88',
d89:'orders.tbl.89',d90:'orders.tbl.90',
d91:'orders.tbl.91',d92:'orders.tbl.92',
d93:'orders.tbl.93',d94:'orders.tbl.94',
d95:'orders.tbl.95',d96:'orders.tbl.96'
))
reject limit unlimited;
drop table ps_et;
create table ps_et(
   ps_partkey        number ,
   ps_suppkey        number ,
   ps_availqty       number ,
   ps_supplycost     number ,
   ps_comment        varchar(199)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
                    records delimited by newline
                    nobadfile
                    nologfile
                    fields terminated by '|'
                    missing field values are null
            )
            location (
d1:'partsupp.tbl.1',d2:'partsupp.tbl.2',
d3:'partsupp.tbl.3',d4:'partsupp.tbl.4',
d5:'partsupp.tbl.5',d6:'partsupp.tbl.6',
d7:'partsupp.tbl.7',d8:'partsupp.tbl.8',
d9:'partsupp.tbl.9',d10:'partsupp.tbl.10',
d11:'partsupp.tbl.11',d12:'partsupp.tbl.12',
d13:'partsupp.tbl.13',d14:'partsupp.tbl.14',
d15:'partsupp.tbl.15',d16:'partsupp.tbl.16',
d17:'partsupp.tbl.17',d18:'partsupp.tbl.18',
d19:'partsupp.tbl.19',d20:'partsupp.tbl.20',
d21:'partsupp.tbl.21',d22:'partsupp.tbl.22',
d23:'partsupp.tbl.23',d24:'partsupp.tbl.24',
d25:'partsupp.tbl.25',d26:'partsupp.tbl.26',
d27:'partsupp.tbl.27',d28:'partsupp.tbl.28',
d29:'partsupp.tbl.29',d30:'partsupp.tbl.30',
d31:'partsupp.tbl.31',d32:'partsupp.tbl.32',
d33:'partsupp.tbl.33',d34:'partsupp.tbl.34',
d35:'partsupp.tbl.35',d36:'partsupp.tbl.36',
d37:'partsupp.tbl.37',d38:'partsupp.tbl.38',
d39:'partsupp.tbl.39',d40:'partsupp.tbl.40',
d41:'partsupp.tbl.41',d42:'partsupp.tbl.42',
d43:'partsupp.tbl.43',d44:'partsupp.tbl.44',
d45:'partsupp.tbl.45',d46:'partsupp.tbl.46',
d47:'partsupp.tbl.47',d48:'partsupp.tbl.48',
d49:'partsupp.tbl.49',d50:'partsupp.tbl.50',
d51:'partsupp.tbl.51',d52:'partsupp.tbl.52',
d53:'partsupp.tbl.53',d54:'partsupp.tbl.54',
d55:'partsupp.tbl.55',d56:'partsupp.tbl.56',
d57:'partsupp.tbl.57',d58:'partsupp.tbl.58',
d59:'partsupp.tbl.59',d60:'partsupp.tbl.60',
d61:'partsupp.tbl.61',d62:'partsupp.tbl.62',
d63:'partsupp.tbl.63',d64:'partsupp.tbl.64',
d65:'partsupp.tbl.65',d66:'partsupp.tbl.66',
d67:'partsupp.tbl.67',d68:'partsupp.tbl.68',
d69:'partsupp.tbl.69',d70:'partsupp.tbl.70',
d71:'partsupp.tbl.71',d72:'partsupp.tbl.72',
d73:'partsupp.tbl.73',d74:'partsupp.tbl.74',
d75:'partsupp.tbl.75',d76:'partsupp.tbl.76',
d77:'partsupp.tbl.77',d78:'partsupp.tbl.78',
d79:'partsupp.tbl.79',d80:'partsupp.tbl.80',
d81:'partsupp.tbl.81',d82:'partsupp.tbl.82',
d83:'partsupp.tbl.83',d84:'partsupp.tbl.84',
d85:'partsupp.tbl.85',d86:'partsupp.tbl.86',
d87:'partsupp.tbl.87',d88:'partsupp.tbl.88',
d89:'partsupp.tbl.89',d90:'partsupp.tbl.90',
d91:'partsupp.tbl.91',d92:'partsupp.tbl.92',
d93:'partsupp.tbl.93',d94:'partsupp.tbl.94',
d95:'partsupp.tbl.95',d96:'partsupp.tbl.96'
))
reject limit unlimited;
drop table p_et;
create table p_et(
   p_partkey         number ,
   p_name            varchar(55) ,
   p_mfgr            char(25) ,
   p_brand           char(10) ,
   p_type            varchar(25) ,
   p_size            number ,
   p_container       char(10) ,
   p_retailprice     number ,
   p_comment         varchar(23)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
                    records delimited by newline
                    nobadfile
                    nologfile
                    fields terminated by '|'
                    missing field values are null
            )
            location (
d1:'part.tbl.1',d2:'part.tbl.2',
d3:'part.tbl.3',d4:'part.tbl.4',
d5:'part.tbl.5',d6:'part.tbl.6',
d7:'part.tbl.7',d8:'part.tbl.8',
d9:'part.tbl.9',d10:'part.tbl.10',
d11:'part.tbl.11',d12:'part.tbl.12',
d13:'part.tbl.13',d14:'part.tbl.14',
d15:'part.tbl.15',d16:'part.tbl.16',
d17:'part.tbl.17',d18:'part.tbl.18',
d19:'part.tbl.19',d20:'part.tbl.20',
d21:'part.tbl.21',d22:'part.tbl.22',
d23:'part.tbl.23',d24:'part.tbl.24',
d25:'part.tbl.25',d26:'part.tbl.26',
d27:'part.tbl.27',d28:'part.tbl.28',
d29:'part.tbl.29',d30:'part.tbl.30',
d31:'part.tbl.31',d32:'part.tbl.32',
d33:'part.tbl.33',d34:'part.tbl.34',
d35:'part.tbl.35',d36:'part.tbl.36',
d37:'part.tbl.37',d38:'part.tbl.38',
d39:'part.tbl.39',d40:'part.tbl.40',
d41:'part.tbl.41',d42:'part.tbl.42',
d43:'part.tbl.43',d44:'part.tbl.44',
d45:'part.tbl.45',d46:'part.tbl.46',
d47:'part.tbl.47',d48:'part.tbl.48',
d49:'part.tbl.49',d50:'part.tbl.50',
d51:'part.tbl.51',d52:'part.tbl.52',
d53:'part.tbl.53',d54:'part.tbl.54',
d55:'part.tbl.55',d56:'part.tbl.56',
d57:'part.tbl.57',d58:'part.tbl.58',
d59:'part.tbl.59',d60:'part.tbl.60',
d61:'part.tbl.61',d62:'part.tbl.62',
```


© 2004 Hewlett Packard Company. All rights reserved

```
d63:'part.tbl.63',d64:'part.tbl.64',
d65:'part.tbl.65',d66:'part.tbl.66',
d67:'part.tbl.67',d68:'part.tbl.68',
d69:'part.tbl.69',d70:'part.tbl.70',
d71:'part.tbl.71',d72:'part.tbl.72',
d73:'part.tbl.73',d74:'part.tbl.74',
d75:'part.tbl.75',d76:'part.tbl.76',
d77:'part.tbl.77',d78:'part.tbl.78',
d79:'part.tbl.79',d80:'part.tbl.80',
d81:'part.tbl.81',d82:'part.tbl.82',
d83:'part.tbl.83',d84:'part.tbl.84',
d85:'part.tbl.85',d86:'part.tbl.86',
d87:'part.tbl.87',d88:'part.tbl.88',
d89:'part.tbl.89',d90:'part.tbl.90',
d91:'part.tbl.91',d92:'part.tbl.92',
d93:'part.tbl.93',d94:'part.tbl.94',
d95:'part.tbl.95',d96:'part.tbl.96'
))
reject limit unlimited;
drop table c_et;
create table c_et(
   c_custkey          number ,
   c_name             varchar(25) ,
   c_address          varchar(40) ,
   c_nationkey        number ,
   c_phone            char(15) ,
   c_acctbal          number ,
   c_mktsegment       char(10) ,
   c_comment          varchar(117)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
                    records delimited by newline
                    nobadfile
                    nologfile
                    fields terminated by '|'
                    missing field values are null
          )
          location (
d1:'customer.tbl.1',d2:'customer.tbl.2',
d3:'customer.tbl.3',d4:'customer.tbl.4',
d5:'customer.tbl.5',d6:'customer.tbl.6',
d7:'customer.tbl.7',d8:'customer.tbl.8',
d9:'customer.tbl.9',d10:'customer.tbl.10',
d11:'customer.tbl.11',d12:'customer.tbl.12',
d13:'customer.tbl.13',d14:'customer.tbl.14',
d15:'customer.tbl.15',d16:'customer.tbl.16',
d17:'customer.tbl.17',d18:'customer.tbl.18',
d19:'customer.tbl.19',d20:'customer.tbl.20',
d21:'customer.tbl.21',d22:'customer.tbl.22',
d23:'customer.tbl.23',d24:'customer.tbl.24',
d25:'customer.tbl.25',d26:'customer.tbl.26',
d27:'customer.tbl.27',d28:'customer.tbl.28',
d29:'customer.tbl.29',d30:'customer.tbl.30',
d31:'customer.tbl.31',d32:'customer.tbl.32',
d33:'customer.tbl.33',d34:'customer.tbl.34',
d35:'customer.tbl.35',d36:'customer.tbl.36',
d37:'customer.tbl.37',d38:'customer.tbl.38',
d39:'customer.tbl.39',d40:'customer.tbl.40',
d41:'customer.tbl.41',d42:'customer.tbl.42',
d43:'customer.tbl.43',d44:'customer.tbl.44',
d45:'customer.tbl.45',d46:'customer.tbl.46',
d47:'customer.tbl.47',d48:'customer.tbl.48',
d49:'customer.tbl.49',d50:'customer.tbl.50',
d51:'customer.tbl.51',d52:'customer.tbl.52',
d53:'customer.tbl.53',d54:'customer.tbl.54',
d55:'customer.tbl.55',d56:'customer.tbl.56',
d57:'customer.tbl.57',d58:'customer.tbl.58',
d59:'customer.tbl.59',d60:'customer.tbl.60',
d61:'customer.tbl.61',d62:'customer.tbl.62',
d63:'customer.tbl.63',d64:'customer.tbl.64',
d65:'customer.tbl.65',d66:'customer.tbl.66',
d67:'customer.tbl.67',d68:'customer.tbl.68',
d69:'customer.tbl.69',d70:'customer.tbl.70',
d71:'customer.tbl.71',d72:'customer.tbl.72',
d73:'customer.tbl.73',d74:'customer.tbl.74',
d75:'customer.tbl.75',d76:'customer.tbl.76',
d77:'customer.tbl.77',d78:'customer.tbl.78',
d79:'customer.tbl.79',d80:'customer.tbl.80',
d81:'customer.tbl.81',d82:'customer.tbl.82',
d83:'customer.tbl.83',d84:'customer.tbl.84',
d85:'customer.tbl.85',d86:'customer.tbl.86',
d87:'customer.tbl.87',d88:'customer.tbl.88',
d89:'customer.tbl.89',d90:'customer.tbl.90',
d91:'customer.tbl.91',d92:'customer.tbl.92',
d93:'customer.tbl.93',d94:'customer.tbl.94',
d95:'customer.tbl.95',d96:'customer.tbl.96'
))
reject limit unlimited;
drop table s_et;
create table s_et(
   s_suppkey          number ,
   s_name             char(25) ,
   s_address          varchar(40) ,
   s_nationkey        number ,
   s_phone            char(15) ,
   s_acctbal          number ,
   s_comment          varchar(101)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
                    records delimited by newline
                    nobadfile
                    nologfile
                    fields terminated by '|'
                    missing field values are null
          )
          location (
d1:'supplier.tbl.1',d2:'supplier.tbl.2',
d3:'supplier.tbl.3',d4:'supplier.tbl.4',
d5:'supplier.tbl.5',d6:'supplier.tbl.6',
d7:'supplier.tbl.7',d8:'supplier.tbl.8',
d9:'supplier.tbl.9',d10:'supplier.tbl.10',
d11:'supplier.tbl.11',d12:'supplier.tbl.12',
d13:'supplier.tbl.13',d14:'supplier.tbl.14',
d15:'supplier.tbl.15',d16:'supplier.tbl.16',
d17:'supplier.tbl.17',d18:'supplier.tbl.18',
d19:'supplier.tbl.19',d20:'supplier.tbl.20',
d21:'supplier.tbl.21',d22:'supplier.tbl.22',
d23:'supplier.tbl.23',d24:'supplier.tbl.24',
d25:'supplier.tbl.25',d26:'supplier.tbl.26',
d27:'supplier.tbl.27',d28:'supplier.tbl.28',
d29:'supplier.tbl.29',d30:'supplier.tbl.30',
d31:'supplier.tbl.31',d32:'supplier.tbl.32',
d33:'supplier.tbl.33',d34:'supplier.tbl.34',
d35:'supplier.tbl.35',d36:'supplier.tbl.36',
d37:'supplier.tbl.37',d38:'supplier.tbl.38',
d39:'supplier.tbl.39',d40:'supplier.tbl.40',
d41:'supplier.tbl.41',d42:'supplier.tbl.42',
d43:'supplier.tbl.43',d44:'supplier.tbl.44',
d45:'supplier.tbl.45',d46:'supplier.tbl.46',
d47:'supplier.tbl.47',d48:'supplier.tbl.48',
d49:'supplier.tbl.49',d50:'supplier.tbl.50',
d51:'supplier.tbl.51',d52:'supplier.tbl.52',
```


© 2004 Hewlett Packard Company. All rights reserved

```
d53:'supplier.tbl.53',d54:'supplier.tbl.54',
d55:'supplier.tbl.55',d56:'supplier.tbl.56',
d57:'supplier.tbl.57',d58:'supplier.tbl.58',
d59:'supplier.tbl.59',d60:'supplier.tbl.60',
d61:'supplier.tbl.61',d62:'supplier.tbl.62',
d63:'supplier.tbl.63',d64:'supplier.tbl.64',
d65:'supplier.tbl.65',d66:'supplier.tbl.66',
d67:'supplier.tbl.67',d68:'supplier.tbl.68',
d69:'supplier.tbl.69',d70:'supplier.tbl.70',
d71:'supplier.tbl.71',d72:'supplier.tbl.72',
d73:'supplier.tbl.73',d74:'supplier.tbl.74',
d75:'supplier.tbl.75',d76:'supplier.tbl.76',
d77:'supplier.tbl.77',d78:'supplier.tbl.78',
d79:'supplier.tbl.79',d80:'supplier.tbl.80',
d81:'supplier.tbl.81',d82:'supplier.tbl.82',
d83:'supplier.tbl.83',d84:'supplier.tbl.84',
d85:'supplier.tbl.85',d86:'supplier.tbl.86',
d87:'supplier.tbl.87',d88:'supplier.tbl.88',
d89:'supplier.tbl.89',d90:'supplier.tbl.90',
d91:'supplier.tbl.91',d92:'supplier.tbl.92',
d93:'supplier.tbl.93',d94:'supplier.tbl.94',
d95:'supplier.tbl.95',d96:'supplier.tbl.96'
            ))
reject limit unlimited;
drop table n_et;
create table n_et(
   n_nationkey        number ,
   n_name             char(25) ,
   n_regionkey        number ,
   n_comment          varchar(152)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
                        records delimited by newline
                        nobadfile
                        nologfile
                        fields terminated by '|'
                        missing field values are null
            )
            location (
            d1:'nation.tbl'))
reject limit unlimited;
drop table r_et;
create table r_et(
   r_regionkey        number ,
   r_name             char(25) ,
   r_comment          varchar(152)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
                        records delimited by newline
                        nobadfile
                        nologfile
                        fields terminated by '|'
                        missing field values are null
            )
            location (
            d1:'region.tbl'))
reject limit unlimited;
alter table l_et parallel 96;
alter table o_et parallel 96;
alter table ps_et parallel 96;
alter table p_et parallel 96;
alter table c_et parallel 96;
alter table s_et parallel 96;
alter user tpch default tablespace ts_default;
alter user tpch temporary tablespace ts_temp;
@?/rdbms/admin/utlxplan.sql;
set timing on
set echo on
!date

-- loading lineitem
--drop table lineitem;
create table lineitem(
   l_shipdate         ,
   l_orderkey         NOT NULL,
   l_discount         NOT NULL,
   l_extendedprice    NOT NULL,
   l_suppkey          NOT NULL,
   l_quantity         NOT NULL,
   l_returnflag       ,
   l_partkey          NOT NULL,
   l_linestatus       ,
   l_tax              NOT NULL,
   l_commitdate       ,
   l_receiptdate      ,
   l_shipmode         ,
   l_linenumber       NOT NULL,
   l_shipinstruct     ,
   l_comment
)
pctfree 1
pctused 99
initrans 10
--storage (initial 100m next 12m freelist groups 8 freelists 99)
storage (initial 80m next 12m freelist groups 12 freelists 99)
compress
parallel 96
nologging
partition by range (l_shipdate)
subpartition by hash (l_partkey)
subpartitions 96
(
partition item1 values less than (to_date('1992-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item2 values less than (to_date('1992-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
```


© 2004 Hewlett Packard Company. All rights reserved

```
partition item3 values less than (to_date('1992-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item4 values less than (to_date('1992-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item5 values less than (to_date('1992-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item6 values less than (to_date('1992-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item7 values less than (to_date('1992-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item8 values less than (to_date('1992-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item9 values less than (to_date('1992-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item10 values less than (to_date('1992-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item11 values less than (to_date('1992-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item12 values less than (to_date('1992-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item13 values less than (to_date('1993-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item14 values less than (to_date('1993-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item15 values less than (to_date('1993-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item16 values less than (to_date('1993-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item17 values less than (to_date('1993-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item18 values less than (to_date('1993-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item19 values less than (to_date('1993-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item20 values less than (to_date('1993-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item21 values less than (to_date('1993-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item22 values less than (to_date('1993-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item23 values less than (to_date('1993-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item24 values less than (to_date('1993-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item25 values less than (to_date('1994-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item26 values less than (to_date('1994-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item27 values less than (to_date('1994-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item28 values less than (to_date('1994-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item29 values less than (to_date('1994-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item30 values less than (to_date('1994-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
```


© 2004 Hewlett Packard Company. All rights reserved

```
partition item31 values less than (to_date('1994-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item32 values less than (to_date('1994-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item33 values less than (to_date('1994-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item34 values less than (to_date('1994-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item35 values less than (to_date('1994-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item36 values less than (to_date('1994-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item37 values less than (to_date('1995-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item38 values less than (to_date('1995-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item39 values less than (to_date('1995-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item40 values less than (to_date('1995-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item41 values less than (to_date('1995-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item42 values less than (to_date('1995-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item43 values less than (to_date('1995-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item44 values less than (to_date('1995-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item45 values less than (to_date('1995-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item46 values less than (to_date('1995-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item47 values less than (to_date('1995-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item48 values less than (to_date('1995-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item49 values less than (to_date('1996-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item50 values less than (to_date('1996-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item51 values less than (to_date('1996-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item52 values less than (to_date('1996-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item53 values less than (to_date('1996-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item54 values less than (to_date('1996-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item55 values less than (to_date('1996-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item56 values less than (to_date('1996-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item57 values less than (to_date('1996-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item58 values less than (to_date('1996-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
```


© 2004 Hewlett Packard Company. All rights reserved

```
partition item59 values less than (to_date('1996-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item60 values less than (to_date('1996-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item61 values less than (to_date('1997-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item62 values less than (to_date('1997-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item63 values less than (to_date('1997-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item64 values less than (to_date('1997-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item65 values less than (to_date('1997-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item66 values less than (to_date('1997-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item67 values less than (to_date('1997-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item68 values less than (to_date('1997-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item69 values less than (to_date('1997-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item70 values less than (to_date('1997-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item71 values less than (to_date('1997-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item72 values less than (to_date('1997-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item73 values less than (to_date('1998-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item74 values less than (to_date('1998-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item75 values less than (to_date('1998-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item76 values less than (to_date('1998-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item77 values less than (to_date('1998-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item78 values less than (to_date('1998-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item79 values less than (to_date('1998-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item80 values less than (to_date('1998-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item81 values less than (to_date('1998-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item82 values less than (to_date('1998-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item83 values less than (to_date('1998-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition item84 values less than (MAXVALUE)
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96))
as select
   l_shipdate        ,
   l_orderkey        ,
   l_discount        ,
   l_extendedprice   ,
   l_suppkey         ,
   l_quantity        ,
   l_returnflag      ,
   l_partkey         ,
   l_linestatus      ,
   l_tax             ,
   l_commitdate      ,
   l_receiptdate     ,
   l_shipmode        ,
   l_linenumber      ,
   l_shipinstruct    ,
   l_comment
from l_et order by l_orderkey;

-- loading orders
!date
rem drop table orders;
create table orders(
   o_orderdate       ,
   o_orderkey        NOT NULL,
   o_custkey         NOT NULL,
   o_orderpriority   ,
   o_shippriority    ,
   o_clerk           ,
   o_orderstatus     ,
   o_totalprice      ,
   o_comment
```


© 2004 Hewlett Packard Company. All rights reserved

```
)
pctfree 1
pctused 99
initrans 10
--storage (initial 35m next 5m freelist groups 8 freelists 99)
storage (initial 28m next 5m freelist groups 12 freelists 99)
compress
parallel 96
nologging
partition by range (o_orderdate)
subpartition by hash (o_custkey)
subpartitions 96
(
partition ord1 values less than (to_date('1992-01-01','YYYY-MM-DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord2 values less than (to_date('1992-02-01','YYYY-MM-DD'))
store in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9,
ts_h10, ts_h11, ts_h12, ts_h13, ts_h14, ts_h15,
ts_h16,ts_h17,ts_h18,ts_h19, ts_h20, ts_h21,ts_h22, ts_h23, ts_h24,
ts_h25, ts_h26, ts_h27, ts_h28, ts_h29, ts_h30,ts_h31, ts_h32, ts_h33,
ts_h34, ts_h35, ts_h36, ts_h37, ts_h38, ts_h39, ts_h40, ts_h41, ts_h42,
ts_h43, ts_h44,ts_h45,ts_h46, ts_h47, ts_h48, ts_h49, ts_h50, ts_h51,
ts_h52, ts_h53, ts_h54, ts_h55, ts_h56, ts_h57, ts_h58,
ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64, ts_h65 ,ts_h66 ,ts_h67
,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73 ,ts_h74 ,ts_h75 ,ts_h76
,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82 ,ts_h83 ,ts_h84 ,ts_h85
,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91 ,ts_h92 ,ts_h93 ,ts_h94
,ts_h95 ,ts_h96)
,
partition ord3 values less than (to_date('1992-03-01','YYYY-MM-DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord4 values less than (to_date('1992-04-01','YYYY-MM-DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord5 values less than (to_date('1992-05-01','YYYY-MM-DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord6 values less than (to_date('1992-06-01','YYYY-MM-DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord7 values less than (to_date('1992-07-01','YYYY-MM-DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord8 values less than (to_date('1992-08-01','YYYY-MM-DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord9 values less than (to_date('1992-09-01','YYYY-MM-DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
```


© 2004 Hewlett Packard Company. All rights reserved

```
partition ord10 values less than (to_date('1992-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord11 values less than (to_date('1992-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord12 values less than (to_date('1992-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord13 values less than (to_date('1993-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord14 values less than (to_date('1993-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord15 values less than (to_date('1993-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord16 values less than (to_date('1993-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord17 values less than (to_date('1993-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord18 values less than (to_date('1993-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord19 values less than (to_date('1993-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord20 values less than (to_date('1993-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord21 values less than (to_date('1993-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord22 values less than (to_date('1993-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord23 values less than (to_date('1993-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord24 values less than (to_date('1993-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord25 values less than (to_date('1994-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord26 values less than (to_date('1994-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord27 values less than (to_date('1994-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord28 values less than (to_date('1994-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord29 values less than (to_date('1994-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord30 values less than (to_date('1994-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord31 values less than (to_date('1994-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord32 values less than (to_date('1994-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord33 values less than (to_date('1994-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord34 values less than (to_date('1994-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord35 values less than (to_date('1994-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord36 values less than (to_date('1994-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord37 values less than (to_date('1995-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
```


© 2004 Hewlett Packard Company. All rights reserved

```
partition ord38 values less than (to_date('1995-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord39 values less than (to_date('1995-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord40 values less than (to_date('1995-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord41 values less than (to_date('1995-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord42 values less than (to_date('1995-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord43 values less than (to_date('1995-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord44 values less than (to_date('1995-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord45 values less than (to_date('1995-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord46 values less than (to_date('1995-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord47 values less than (to_date('1995-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord48 values less than (to_date('1995-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord49 values less than (to_date('1996-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord50 values less than (to_date('1996-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord51 values less than (to_date('1996-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord52 values less than (to_date('1996-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord53 values less than (to_date('1996-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord54 values less than (to_date('1996-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord55 values less than (to_date('1996-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord56 values less than (to_date('1996-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord57 values less than (to_date('1996-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord58 values less than (to_date('1996-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord59 values less than (to_date('1996-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord60 values less than (to_date('1996-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord61 values less than (to_date('1997-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord62 values less than (to_date('1997-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord63 values less than (to_date('1997-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord64 values less than (to_date('1997-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord65 values less than (to_date('1997-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
```


© 2004 Hewlett Packard Company. All rights reserved

```
partition ord66 values less than (to_date('1997-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord67 values less than (to_date('1997-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord68 values less than (to_date('1997-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord69 values less than (to_date('1997-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord70 values less than (to_date('1997-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord71 values less than (to_date('1997-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord72 values less than (to_date('1997-12-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord73 values less than (to_date('1998-01-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord74 values less than (to_date('1998-02-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord75 values less than (to_date('1998-03-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
```


© 2004 Hewlett Packard Company. All rights reserved

```
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord76 values less than (to_date('1998-04-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord77 values less than (to_date('1998-05-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord78 values less than (to_date('1998-06-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord79 values less than (to_date('1998-07-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord80 values less than (to_date('1998-08-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord81 values less than (to_date('1998-09-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord82 values less than (to_date('1998-10-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord83 values less than (to_date('1998-11-01','YYYY-MM-
DD'))
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
,
partition ord84 values less than (MAXVALUE)
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
```


© 2004 Hewlett Packard Company. All rights reserved

```
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
)
as select
   o_orderdate        ,
   o_orderkey         ,
   o_custkey          ,
   o_orderpriority    ,
   o_shippriority     ,
   o_clerk            ,
   o_orderstatus      ,
   o_totalprice       ,
   o_comment
from o_et order by o_orderkey;
!date

-- loading partsupp
!date
rem drop table partsupp;
create table partsupp(
   ps_partkey         NOT NULL,
   ps_suppkey         NOT NULL,
   ps_supplycost      NOT NULL,
   ps_availqty        ,
   ps_comment         ,
constraint pk_partkey_suppkey_1 primary key(ps_partkey, ps_suppkey)
)
organization index
partition by hash (ps_partkey)
partitions 96
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
storage (initial 250m next 80m)
parallel 96
nologging
pctthreshold 50
as select
   ps_partkey         ,
   ps_suppkey         ,
   ps_supplycost      ,
   ps_availqty        ,
   ps_comment
from ps_et;
!date
rem drop table customer;
create table customer(
   c_custkey          NOT NULL,
   c_mktsegment       ,
   c_nationkey        ,
   c_name             ,
   c_address          ,
   c_phone            ,
   c_acctbal          ,
   c_comment
)
pctfree 0
pctused 99
parallel 96
nologging
--storage (initial 180m next 6m)
storage (initial 120m next 6m)
compress
partition by hash (c_custkey)
partitions 96
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
as select
   c_custkey          ,
   c_mktsegment       ,
   c_nationkey        ,
   c_name             ,
   c_address          ,
   c_phone            ,
   c_acctbal          ,
   c_comment
from c_et;
!date
rem drop table part;

create table part(
   p_partkey          NOT NULL,
   p_type             ,
   p_size             ,
   p_brand            ,
   p_name             ,
   p_container        ,
   p_mfgr             ,
   p_retailprice      ,
   p_comment
)
pctfree 0
pctused 99
parallel 96
nologging
--storage (initial 200m next 6m)
storage (initial 160m next 6m)
compress
partition by hash (p_partkey)
partitions 96
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
as select
   p_partkey          ,
   p_type             ,
   p_size             ,
   p_brand            ,
   p_name             ,
   p_container        ,
   p_mfgr             ,
```


© 2004 Hewlett Packard Company. All rights reserved

```
   p_retailprice        ,
   p_comment
from p_et;
!date
rem drop table supplier;
create table supplier(
   s_suppkey            NOT NULL,
   s_nationkey          ,
   s_comment            ,
   s_name               ,
   s_address            ,
   s_phone              ,
   s_acctbal
)
pctfree 0
pctused 99
parallel 96
nologging
--storage (initial 20m next 2m)
storage (initial 18m next 2m)
compress
partition by hash (s_suppkey)
partitions 96
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
as select
   s_suppkey            ,
   s_nationkey          ,
   s_comment            ,
   s_name               ,
   s_address            ,
   s_phone              ,
   s_acctbal
from s_et;

rem drop table nation;
create table nation(
   n_nationkey          NOT NULL,
   n_name               ,
   n_regionkey          ,
   n_comment            )
tablespace ts_default
as select * from n_et;

rem drop table region;
create table region(
   r_regionkey          ,
   r_name               ,
   r_comment            )
tablespace ts_default
as select * from r_et;
!date
drop table l_et;
drop table o_et;
drop table ps_et;
drop table p_et;
drop table c_et;
drop table s_et;
drop table n_et;
drop table r_et;
exit;
!
```

ixcre.sh

```
#!/bin/ksh
echo Index Creation Phase
sqlplus /NOLOG <<!
connect tpch/tpch;
!date
set echo on
set timing on
drop index i_l_orderkey;
create index i_l_orderkey
on lineitem (l_orderkey) global partition by hash (l_orderkey)
partitions 96
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
pctfree 2
initrans 10
storage (initial 192m next 192m pctincrease 0 freelist groups 12 freelists
99)
parallel 96
compute statistics
nologging;
alter index i_l_orderkey allocate extent (size 256m);
alter index i_l_orderkey allocate extent (size 256m);
alter index i_l_orderkey allocate extent (size 256m);
alter index i_l_orderkey allocate extent (size 256m);
!date
set echo on
set timing on
drop index i_o_orderkey;
create unique index i_o_orderkey
on orders (o_orderkey) global partition by hash (o_orderkey)
partitions 96
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
pctfree 2
initrans 10
storage (initial 192m next 192m pctincrease 0 freelist groups 12 freelists
99 )
parallel 96
compute statistics
nologging;
alter index i_o_orderkey allocate extent (size 256m);
alter index i_o_orderkey allocate extent (size 256m);
!date
set echo on
set timing on
```


© 2004 Hewlett Packard Company. All rights reserved

```
--drop index i_c_custkey;
create unique index i_c_custkey
on customer (c_custkey) global partition by hash (c_custkey)
partitions 96
store
in (ts_h1, ts_h2, ts_h3, ts_h4, ts_h5, ts_h6, ts_h7, ts_h8, ts_h9, ts_h10,
ts_h11, ts_h12, ts_h13, ts_h14, ts_h15, ts_h16,ts_h17,ts_h18,ts_h19,
ts_h20, ts_h21,ts_h22, ts_h23, ts_h24, ts_h25, ts_h26, ts_h27, ts_h28,
ts_h29, ts_h30,ts_h31, ts_h32, ts_h33, ts_h34, ts_h35, ts_h36, ts_h37,
ts_h38, ts_h39, ts_h40, ts_h41, ts_h42, ts_h43, ts_h44,ts_h45,ts_h46,
ts_h47, ts_h48, ts_h49, ts_h50, ts_h51, ts_h52, ts_h53, ts_h54, ts_h55,
ts_h56, ts_h57, ts_h58, ts_h59,ts_h60,ts_h61, ts_h62, ts_h63, ts_h64,
ts_h65 ,ts_h66 ,ts_h67 ,ts_h68 ,ts_h69 ,ts_h70 ,ts_h71 ,ts_h72 ,ts_h73
,ts_h74 ,ts_h75 ,ts_h76 ,ts_h77 ,ts_h78 ,ts_h79 ,ts_h80 ,ts_h81 ,ts_h82
,ts_h83 ,ts_h84 ,ts_h85 ,ts_h86 ,ts_h87 ,ts_h88 ,ts_h89 ,ts_h90 ,ts_h91
,ts_h92 ,ts_h93 ,ts_h94 ,ts_h95 ,ts_h96)
pctfree 2
initrans 10
storage (initial 192m next 192m pctincrease 0 freelist groups 12 freelists
99 )
parallel 96
compute statistics
nologging;
!date
exit;
!
```

---
backup.sh

```
date >> backup.log
for a in 2 3 4 5
do
 let b=${a}-1
 rsh h${a} umount /home/oracle/dev/ff*
 rsh h${a} /root/backup/mntbackup${b}.sh
 rsh h${a} /root/backup/backup${b}.sh &
done
wait
date >> backup.log
```

---
mntbackup1.sh

```
mount /dev/sdb6 /home/oracle/backup/backup_1
mount /dev/sdc6 /home/oracle/backup/backup_2
mount /dev/sdd6 /home/oracle/backup/backup_3
mount /dev/sde6 /home/oracle/backup/backup_4
mount /dev/sdf6 /home/oracle/backup/backup_5
mount /dev/sdg6 /home/oracle/backup/backup_6
mount /dev/sdh6 /home/oracle/backup/backup_7
mount /dev/sdi6 /home/oracle/backup/backup_8
mount /dev/sdk6 /home/oracle/backup/backup_9
mount /dev/sdl6 /home/oracle/backup/backup_10
mount /dev/sdm6 /home/oracle/backup/backup_11
mount /dev/sdn6 /home/oracle/backup/backup_12
mount /dev/sdo6 /home/oracle/backup/backup_13
mount /dev/sdp6 /home/oracle/backup/backup_14
mount /dev/sdq6 /home/oracle/backup/backup_15
mount /dev/sdr6 /home/oracle/backup/backup_16
mount /dev/sdt6 /home/oracle/backup/backup_17
mount /dev/sdu6 /home/oracle/backup/backup_18
mount /dev/sdv6 /home/oracle/backup/backup_19
mount /dev/sdw6 /home/oracle/backup/backup_20
mount /dev/sdx6 /home/oracle/backup/backup_21
mount /dev/sdy6 /home/oracle/backup/backup_22
mount /dev/sdz6 /home/oracle/backup/backup_23
mount /dev/sdaa6 /home/oracle/backup/backup_24
```

---
mntbackup2.sh

```
mount /dev/sdac6 /home/oracle/backup/backup_25
mount /dev/sdad6 /home/oracle/backup/backup_26
mount /dev/sdae6 /home/oracle/backup/backup_27
mount /dev/sdaf6 /home/oracle/backup/backup_28
mount /dev/sdag6 /home/oracle/backup/backup_29
mount /dev/sdah6 /home/oracle/backup/backup_30
mount /dev/sdai6 /home/oracle/backup/backup_31
mount /dev/sdaj6 /home/oracle/backup/backup_32
mount /dev/sdal6 /home/oracle/backup/backup_33
mount /dev/sdam6 /home/oracle/backup/backup_34
mount /dev/sdan6 /home/oracle/backup/backup_35
mount /dev/sdao6 /home/oracle/backup/backup_36
mount /dev/sdap6 /home/oracle/backup/backup_37
mount /dev/sdaq6 /home/oracle/backup/backup_38
mount /dev/sdar6 /home/oracle/backup/backup_39
mount /dev/sdas6 /home/oracle/backup/backup_40
mount /dev/sdau6 /home/oracle/backup/backup_41
mount /dev/sdav6 /home/oracle/backup/backup_42
mount /dev/sdaw6 /home/oracle/backup/backup_43
mount /dev/sdax6 /home/oracle/backup/backup_44
mount /dev/sday6 /home/oracle/backup/backup_45
mount /dev/sdaz6 /home/oracle/backup/backup_46
mount /dev/sdba6 /home/oracle/backup/backup_47
mount /dev/sdbb6 /home/oracle/backup/backup_48
```

---
mntbackup3.sh

```
mount /dev/sdbd6 /home/oracle/backup/backup_49
mount /dev/sdbe6 /home/oracle/backup/backup_50
mount /dev/sdbf6 /home/oracle/backup/backup_51
mount /dev/sdbg6 /home/oracle/backup/backup_52
mount /dev/sdbh6 /home/oracle/backup/backup_53
mount /dev/sdbi6 /home/oracle/backup/backup_54
mount /dev/sdbj6 /home/oracle/backup/backup_55
mount /dev/sdbk6 /home/oracle/backup/backup_56
mount /dev/sdbm6 /home/oracle/backup/backup_57
mount /dev/sdbn6 /home/oracle/backup/backup_58
mount /dev/sdbo6 /home/oracle/backup/backup_59
mount /dev/sdbp6 /home/oracle/backup/backup_60
mount /dev/sdbq6 /home/oracle/backup/backup_61
mount /dev/sdbr6 /home/oracle/backup/backup_62
mount /dev/sdbs6 /home/oracle/backup/backup_63
mount /dev/sdbt6 /home/oracle/backup/backup_64
mount /dev/sdbv6 /home/oracle/backup/backup_65
mount /dev/sdbw6 /home/oracle/backup/backup_66
mount /dev/sdbx6 /home/oracle/backup/backup_67
mount /dev/sdby6 /home/oracle/backup/backup_68
mount /dev/sdbz6 /home/oracle/backup/backup_69
mount /dev/sdca6 /home/oracle/backup/backup_70
mount /dev/sdcb6 /home/oracle/backup/backup_71
mount /dev/sdcc6 /home/oracle/backup/backup_72
```

---
mntbackup4.sh

```
mount /dev/sdce6 /home/oracle/backup/backup_73
mount /dev/sdcf6 /home/oracle/backup/backup_74
mount /dev/sdcg6 /home/oracle/backup/backup_75
mount /dev/sdch6 /home/oracle/backup/backup_76
mount /dev/sdci6 /home/oracle/backup/backup_77
mount /dev/sdcj6 /home/oracle/backup/backup_78
mount /dev/sdck6 /home/oracle/backup/backup_79
mount /dev/sdcl6 /home/oracle/backup/backup_80
mount /dev/sdcn6 /home/oracle/backup/backup_81
mount /dev/sdco6 /home/oracle/backup/backup_82
mount /dev/sdcp6 /home/oracle/backup/backup_83
```


© 2004 Hewlett Packard Company. All rights reserved

```
mount /dev/sdcq6 /home/oracle/backup/backup_84
mount /dev/sdcr6 /home/oracle/backup/backup_85
mount /dev/sdcs6 /home/oracle/backup/backup_86
mount /dev/sdct6 /home/oracle/backup/backup_87
mount /dev/sdcu6 /home/oracle/backup/backup_88
mount /dev/sdcw6 /home/oracle/backup/backup_89
mount /dev/sdcx6 /home/oracle/backup/backup_90
mount /dev/sdcy6 /home/oracle/backup/backup_91
mount /dev/sdcz6 /home/oracle/backup/backup_92
mount /dev/sdda6 /home/oracle/backup/backup_93
mount /dev/sddb6 /home/oracle/backup/backup_94
mount /dev/sddc6 /home/oracle/backup/backup_95
mount /dev/sddd6 /home/oracle/backup/backup_96

----------------------------------------
backup1.sh
----------------------------------------
date > backup1.log
{
dd of=/home/oracle/backup/backup_8/lo_1
if=/home/oracle/dev/raw/lo_1 bs=1024k count=19001
dd of=/home/oracle/backup/backup_8/t_1 if=/home/oracle/dev/raw/t_1
bs=1024k count=12000
dd of=/home/oracle/backup/backup_8/log_1_1
if=/home/oracle/dev/raw/log_1_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_7/lo_2
if=/home/oracle/dev/raw/lo_2 bs=1024k count=19001
dd of=/home/oracle/backup/backup_7/t_2 if=/home/oracle/dev/raw/t_2
bs=1024k count=12000
dd of=/home/oracle/backup/backup_7/sys_1
if=/home/oracle/dev/raw/sys_1 bs=1024k count=1200
} &

{
dd of=/home/oracle/backup/backup_6/lo_3
if=/home/oracle/dev/raw/lo_3 bs=1024k count=19001
dd of=/home/oracle/backup/backup_6/t_3 if=/home/oracle/dev/raw/t_3
bs=1024k count=12000
dd of=/home/oracle/backup/backup_6/undo_1
if=/home/oracle/dev/raw/undo_1 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_5/lo_4
if=/home/oracle/dev/raw/lo_4 bs=1024k count=19001
dd of=/home/oracle/backup/backup_5/t_4 if=/home/oracle/dev/raw/t_4
bs=1024k count=12000
dd of=/home/oracle/backup/backup_5/log_1_2
if=/home/oracle/dev/raw/log_1_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_4/lo_5
if=/home/oracle/dev/raw/lo_5 bs=1024k count=19001
dd of=/home/oracle/backup/backup_4/t_5 if=/home/oracle/dev/raw/t_5
bs=1024k count=12000
dd of=/home/oracle/backup/backup_4/ocr_1
if=/home/oracle/dev/raw/ocr_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_3/lo_6
if=/home/oracle/dev/raw/lo_6 bs=1024k count=19001
dd of=/home/oracle/backup/backup_3/t_6 if=/home/oracle/dev/raw/t_6
bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_2/lo_7
if=/home/oracle/dev/raw/lo_7 bs=1024k count=19001
dd of=/home/oracle/backup/backup_2/t_7 if=/home/oracle/dev/raw/t_7
bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_1/lo_8
if=/home/oracle/dev/raw/lo_8 bs=1024k count=19001
dd of=/home/oracle/backup/backup_1/t_8 if=/home/oracle/dev/raw/t_8
bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_16/lo_9
if=/home/oracle/dev/raw/lo_9 bs=1024k count=19001
dd of=/home/oracle/backup/backup_16/t_9 if=/home/oracle/dev/raw/t_9
bs=1024k count=12000
dd of=/home/oracle/backup/backup_16/quorum_1
if=/home/oracle/dev/raw/quorum_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_15/lo_10
if=/home/oracle/dev/raw/lo_10 bs=1024k count=19001
dd of=/home/oracle/backup/backup_15/t_10
if=/home/oracle/dev/raw/t_10 bs=1024k count=12000
dd of=/home/oracle/backup/backup_15/log_2_1
if=/home/oracle/dev/raw/log_2_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_14/lo_11
if=/home/oracle/dev/raw/lo_11 bs=1024k count=19001
dd of=/home/oracle/backup/backup_14/t_11
if=/home/oracle/dev/raw/t_11 bs=1024k count=12000
dd of=/home/oracle/backup/backup_14/undo_2
if=/home/oracle/dev/raw/undo_2 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_13/lo_12
if=/home/oracle/dev/raw/lo_12 bs=1024k count=19001
dd of=/home/oracle/backup/backup_13/t_12
if=/home/oracle/dev/raw/t_12 bs=1024k count=12000
dd of=/home/oracle/backup/backup_13/log_2_2
if=/home/oracle/dev/raw/log_2_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_12/lo_13
if=/home/oracle/dev/raw/lo_13 bs=1024k count=19001
dd of=/home/oracle/backup/backup_12/t_13
if=/home/oracle/dev/raw/t_13 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_11/lo_14
if=/home/oracle/dev/raw/lo_14 bs=1024k count=19001
dd of=/home/oracle/backup/backup_11/t_14
if=/home/oracle/dev/raw/t_14 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_10/lo_15
if=/home/oracle/dev/raw/lo_15 bs=1024k count=19001
dd of=/home/oracle/backup/backup_10/t_15
if=/home/oracle/dev/raw/t_15 bs=1024k count=12000
} &
```


© 2004 Hewlett Packard Company. All rights reserved

```
{
dd of=/home/oracle/backup/backup_9/lo_16
if=/home/oracle/dev/raw/lo_16 bs=1024k count=19001
dd of=/home/oracle/backup/backup_9/t_16
if=/home/oracle/dev/raw/t_16 bs=1024k count=12000
} &


{
dd of=/home/oracle/backup/backup_24/lo_17
if=/home/oracle/dev/raw/lo_17 bs=1024k count=19001
dd of=/home/oracle/backup/backup_24/t_17
if=/home/oracle/dev/raw/t_17 bs=1024k count=12000
dd of=/home/oracle/backup/backup_24/log_3_1
if=/home/oracle/dev/raw/log_3_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_23/lo_18
if=/home/oracle/dev/raw/lo_18 bs=1024k count=19001
dd of=/home/oracle/backup/backup_23/t_18
if=/home/oracle/dev/raw/t_18 bs=1024k count=12000
dd of=/home/oracle/backup/backup_23/perf_1
if=/home/oracle/dev/raw/perf_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_22/lo_19
if=/home/oracle/dev/raw/lo_19 bs=1024k count=19001
dd of=/home/oracle/backup/backup_22/t_19
if=/home/oracle/dev/raw/t_19 bs=1024k count=12000
dd of=/home/oracle/backup/backup_22/undo_3
if=/home/oracle/dev/raw/undo_3 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_21/lo_20
if=/home/oracle/dev/raw/lo_20 bs=1024k count=19001
dd of=/home/oracle/backup/backup_21/t_20
if=/home/oracle/dev/raw/t_20 bs=1024k count=12000
dd of=/home/oracle/backup/backup_21/log_3_2
if=/home/oracle/dev/raw/log_3_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_20/lo_21
if=/home/oracle/dev/raw/lo_21 bs=1024k count=19001
dd of=/home/oracle/backup/backup_20/t_21
if=/home/oracle/dev/raw/t_21 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_19/lo_22
if=/home/oracle/dev/raw/lo_22 bs=1024k count=19001
dd of=/home/oracle/backup/backup_19/t_22
if=/home/oracle/dev/raw/t_22 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_18/lo_23
if=/home/oracle/dev/raw/lo_23 bs=1024k count=19001
dd of=/home/oracle/backup/backup_18/t_23
if=/home/oracle/dev/raw/t_23 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_17/lo_24
if=/home/oracle/dev/raw/lo_24 bs=1024k count=19001
dd of=/home/oracle/backup/backup_17/t_24
if=/home/oracle/dev/raw/t_24 bs=1024k count=12000
} &
wait
date >> backup1.log

----------------------------------------
backup2.sh
----------------------------------------
date > backup2.log
{
dd of=/home/oracle/backup/backup_32/lo_25
if=/home/oracle/dev/raw/lo_25 bs=1024k count=19001
dd of=/home/oracle/backup/backup_32/t_25
if=/home/oracle/dev/raw/t_25 bs=1024k count=12000
dd of=/home/oracle/backup/backup_32/log_4_1
if=/home/oracle/dev/raw/log_4_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_31/lo_26
if=/home/oracle/dev/raw/lo_26 bs=1024k count=19001
dd of=/home/oracle/backup/backup_31/t_26
if=/home/oracle/dev/raw/t_26 bs=1024k count=12000
dd of=/home/oracle/backup/backup_31/sysaux_1
if=/home/oracle/dev/raw/sysaux_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_30/lo_27
if=/home/oracle/dev/raw/lo_27 bs=1024k count=19001
dd of=/home/oracle/backup/backup_30/t_27
if=/home/oracle/dev/raw/t_27 bs=1024k count=12000
dd of=/home/oracle/backup/backup_30/undo_4
if=/home/oracle/dev/raw/undo_4 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_29/lo_28
if=/home/oracle/dev/raw/lo_28 bs=1024k count=19001
dd of=/home/oracle/backup/backup_29/t_28
if=/home/oracle/dev/raw/t_28 bs=1024k count=12000
dd of=/home/oracle/backup/backup_29/log_4_2
if=/home/oracle/dev/raw/log_4_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_28/lo_29
if=/home/oracle/dev/raw/lo_29 bs=1024k count=19001
dd of=/home/oracle/backup/backup_28/t_29
if=/home/oracle/dev/raw/t_29 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_27/lo_30
if=/home/oracle/dev/raw/lo_30 bs=1024k count=19001
dd of=/home/oracle/backup/backup_27/t_30
if=/home/oracle/dev/raw/t_30 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_26/lo_31
if=/home/oracle/dev/raw/lo_31 bs=1024k count=19001
dd of=/home/oracle/backup/backup_26/t_31
if=/home/oracle/dev/raw/t_31 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_25/lo_32
if=/home/oracle/dev/raw/lo_32 bs=1024k count=19001
```


© 2004 Hewlett Packard Company. All rights reserved

```
dd of=/home/oracle/backup/backup_25/t_32
if=/home/oracle/dev/raw/t_32 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_40/lo_33
if=/home/oracle/dev/raw/lo_33 bs=1024k count=19001
dd of=/home/oracle/backup/backup_40/t_33
if=/home/oracle/dev/raw/t_33 bs=1024k count=12000
dd of=/home/oracle/backup/backup_40/log_5_1
if=/home/oracle/dev/raw/log_5_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_39/lo_34
if=/home/oracle/dev/raw/lo_34 bs=1024k count=19001
dd of=/home/oracle/backup/backup_39/t_34
if=/home/oracle/dev/raw/t_34 bs=1024k count=12000
dd of=/home/oracle/backup/backup_39/default_1
if=/home/oracle/dev/raw/default_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_38/lo_35
if=/home/oracle/dev/raw/lo_35 bs=1024k count=19001
dd of=/home/oracle/backup/backup_38/t_35
if=/home/oracle/dev/raw/t_35 bs=1024k count=12000
dd of=/home/oracle/backup/backup_38/undo_5
if=/home/oracle/dev/raw/undo_5 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_37/lo_36
if=/home/oracle/dev/raw/lo_36 bs=1024k count=19001
dd of=/home/oracle/backup/backup_37/t_36
if=/home/oracle/dev/raw/t_36 bs=1024k count=12000
dd of=/home/oracle/backup/backup_37/log_5_2
if=/home/oracle/dev/raw/log_5_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_36/lo_37
if=/home/oracle/dev/raw/lo_37 bs=1024k count=19001
dd of=/home/oracle/backup/backup_36/t_37
if=/home/oracle/dev/raw/t_37 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_35/lo_38
if=/home/oracle/dev/raw/lo_38 bs=1024k count=19001
dd of=/home/oracle/backup/backup_35/t_38
if=/home/oracle/dev/raw/t_38 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_34/lo_39
if=/home/oracle/dev/raw/lo_39 bs=1024k count=19001
dd of=/home/oracle/backup/backup_34/t_39
if=/home/oracle/dev/raw/t_39 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_33/lo_40
if=/home/oracle/dev/raw/lo_40 bs=1024k count=19001
dd of=/home/oracle/backup/backup_33/t_40
if=/home/oracle/dev/raw/t_40 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_48/lo_41
if=/home/oracle/dev/raw/lo_41 bs=1024k count=19001
dd of=/home/oracle/backup/backup_48/t_41
if=/home/oracle/dev/raw/t_41 bs=1024k count=12000
dd of=/home/oracle/backup/backup_48/log_6_1
if=/home/oracle/dev/raw/log_6_1 bs=1024k
} &

{
dd of=/home/oracle/backup/backup_47/lo_42
if=/home/oracle/dev/raw/lo_42 bs=1024k count=19001
dd of=/home/oracle/backup/backup_47/t_42
if=/home/oracle/dev/raw/t_42 bs=1024k count=12000
dd of=/home/oracle/backup/backup_47/control_001
if=/home/oracle/dev/raw/control_001 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_46/lo_43
if=/home/oracle/dev/raw/lo_43 bs=1024k count=19001
dd of=/home/oracle/backup/backup_46/t_43
if=/home/oracle/dev/raw/t_43 bs=1024k count=12000
dd of=/home/oracle/backup/backup_46/undo_6
if=/home/oracle/dev/raw/undo_6 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_45/lo_44
if=/home/oracle/dev/raw/lo_44 bs=1024k count=19001
dd of=/home/oracle/backup/backup_45/t_44
if=/home/oracle/dev/raw/t_44 bs=1024k count=12000
dd of=/home/oracle/backup/backup_45/log_6_2
if=/home/oracle/dev/raw/log_6_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_44/lo_45
if=/home/oracle/dev/raw/lo_45 bs=1024k count=19001
dd of=/home/oracle/backup/backup_44/t_45
if=/home/oracle/dev/raw/t_45 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_43/lo_46
if=/home/oracle/dev/raw/lo_46 bs=1024k count=19001
dd of=/home/oracle/backup/backup_43/t_46
if=/home/oracle/dev/raw/t_46 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_42/lo_47
if=/home/oracle/dev/raw/lo_47 bs=1024k count=19001
dd of=/home/oracle/backup/backup_42/t_47
if=/home/oracle/dev/raw/t_47 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_41/lo_48
if=/home/oracle/dev/raw/lo_48 bs=1024k count=19001
dd of=/home/oracle/backup/backup_41/t_48
if=/home/oracle/dev/raw/t_48 bs=1024k count=12000
} &
wait
date >> backup2.log

----------
backup3.sh
----------
date > backup3.log
{
```


© 2004 Hewlett Packard Company. All rights reserved

```
dd of=/home/oracle/backup/backup_56/lo_49
if=/home/oracle/dev/raw/lo_49 bs=1024k count=19001
dd of=/home/oracle/backup/backup_56/t_49
if=/home/oracle/dev/raw/t_49 bs=1024k count=12000
dd of=/home/oracle/backup/backup_56/log_7_1
if=/home/oracle/dev/raw/log_7_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_55/lo_50
if=/home/oracle/dev/raw/lo_50 bs=1024k count=19001
dd of=/home/oracle/backup/backup_55/t_50
if=/home/oracle/dev/raw/t_50 bs=1024k count=12000
dd of=/home/oracle/backup/backup_55/control_002
if=/home/oracle/dev/raw/control_002 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_54/lo_51
if=/home/oracle/dev/raw/lo_51 bs=1024k count=19001
dd of=/home/oracle/backup/backup_54/t_51
if=/home/oracle/dev/raw/t_51 bs=1024kc count=12000
dd of=/home/oracle/backup/backup_54/undo_7
if=/home/oracle/dev/raw/undo_7 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_53/lo_52
if=/home/oracle/dev/raw/lo_52 bs=1024k count=19001
dd of=/home/oracle/backup/backup_53/t_52
if=/home/oracle/dev/raw/t_52 bs=1024k count=12000
dd of=/home/oracle/backup/backup_53/log_7_2
if=/home/oracle/dev/raw/log_7_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_52/lo_53
if=/home/oracle/dev/raw/lo_53 bs=1024k count=19001
dd of=/home/oracle/backup/backup_52/t_53
if=/home/oracle/dev/raw/t_53 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_51/lo_54
if=/home/oracle/dev/raw/lo_54 bs=1024k count=19001
dd of=/home/oracle/backup/backup_51/t_54
if=/home/oracle/dev/raw/t_54 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_50/lo_55
if=/home/oracle/dev/raw/lo_55 bs=1024k count=19001
dd of=/home/oracle/backup/backup_50/t_55
if=/home/oracle/dev/raw/t_55 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_49/lo_56
if=/home/oracle/dev/raw/lo_56 bs=1024k count=19001
dd of=/home/oracle/backup/backup_49/t_56
if=/home/oracle/dev/raw/t_56 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_64/lo_57
if=/home/oracle/dev/raw/lo_57 bs=1024k count=19001
dd of=/home/oracle/backup/backup_64/t_57
if=/home/oracle/dev/raw/t_57 bs=1024k count=12000
dd of=/home/oracle/backup/backup_64/log_8_1
if=/home/oracle/dev/raw/log_8_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_63/lo_58
if=/home/oracle/dev/raw/lo_58 bs=1024k count=19001
dd of=/home/oracle/backup/backup_63/t_58
if=/home/oracle/dev/raw/t_58 bs=1024k count=12000
dd of=/home/oracle/backup/backup_63/log_8_2
if=/home/oracle/dev/raw/log_8_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_62/lo_59
if=/home/oracle/dev/raw/lo_59 bs=1024k count=19001
dd of=/home/oracle/backup/backup_62/t_59
if=/home/oracle/dev/raw/t_59 bs=1024k count=12000
dd of=/home/oracle/backup/backup_62/undo_8
if=/home/oracle/dev/raw/undo_8 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_61/lo_60
if=/home/oracle/dev/raw/lo_60 bs=1024k count=19001
dd of=/home/oracle/backup/backup_61/t_60
if=/home/oracle/dev/raw/t_60 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_60/lo_61
if=/home/oracle/dev/raw/lo_61 bs=1024k count=19001
dd of=/home/oracle/backup/backup_60/t_61
if=/home/oracle/dev/raw/t_61 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_59/lo_62
if=/home/oracle/dev/raw/lo_62 bs=1024k count=19001
dd of=/home/oracle/backup/backup_59/t_62
if=/home/oracle/dev/raw/t_62 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_58/lo_63
if=/home/oracle/dev/raw/lo_63 bs=1024k count=19001
dd of=/home/oracle/backup/backup_58/t_63
if=/home/oracle/dev/raw/t_63 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_57/lo_64
if=/home/oracle/dev/raw/lo_64 bs=1024k count=19001
dd of=/home/oracle/backup/backup_57/t_64
if=/home/oracle/dev/raw/t_64 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_72/lo_65
if=/home/oracle/dev/raw/lo_65 bs=1024k count=19001
dd of=/home/oracle/backup/backup_72/t_65
if=/home/oracle/dev/raw/t_65 bs=1024k count=12000
dd of=/home/oracle/backup/backup_72/log_9_1
if=/home/oracle/dev/raw/log_9_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_71/lo_66
if=/home/oracle/dev/raw/lo_66 bs=1024k count=19001
dd of=/home/oracle/backup/backup_71/t_66
if=/home/oracle/dev/raw/t_66 bs=1024k count=12000
```


© 2004 Hewlett Packard Company. All rights reserved

```
dd of=/home/oracle/backup/backup_71/log_9_2
if=/home/oracle/dev/raw/log_9_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_70/lo_67
if=/home/oracle/dev/raw/lo_67 bs=1024k count=19001
dd of=/home/oracle/backup/backup_70/t_67
if=/home/oracle/dev/raw/t_67 bs=1024k count=12000
dd of=/home/oracle/backup/backup_70/undo_9
if=/home/oracle/dev/raw/undo_9 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_69/lo_68
if=/home/oracle/dev/raw/lo_68 bs=1024k count=19001
dd of=/home/oracle/backup/backup_69/t_68
if=/home/oracle/dev/raw/t_68 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_68/lo_69
if=/home/oracle/dev/raw/lo_69 bs=1024k count=19001
dd of=/home/oracle/backup/backup_68/t_69
if=/home/oracle/dev/raw/t_69 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_67/lo_70
if=/home/oracle/dev/raw/lo_70 bs=1024k count=19001
dd of=/home/oracle/backup/backup_67/t_70
if=/home/oracle/dev/raw/t_70 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_66/lo_71
if=/home/oracle/dev/raw/lo_71 bs=1024k count=19001
dd of=/home/oracle/backup/backup_66/t_71
if=/home/oracle/dev/raw/t_71 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_65/lo_72
if=/home/oracle/dev/raw/lo_72 bs=1024k count=19001
dd of=/home/oracle/backup/backup_65/t_72
if=/home/oracle/dev/raw/t_72 bs=1024k count=12000
} &
wait
date >> backup3.log

----------------------------------------
backup4.sh
----------------------------------------
date > backup4.log
{
dd of=/home/oracle/backup/backup_80/lo_73
if=/home/oracle/dev/raw/lo_73 bs=1024k count=19001
dd of=/home/oracle/backup/backup_80/t_73
if=/home/oracle/dev/raw/t_73 bs=1024k count=12000
dd of=/home/oracle/backup/backup_80/log_10_1
if=/home/oracle/dev/raw/log_10_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_79/lo_74
if=/home/oracle/dev/raw/lo_74 bs=1024k count=19001
dd of=/home/oracle/backup/backup_79/t_74
if=/home/oracle/dev/raw/t_74 bs=1024k count=12000
dd of=/home/oracle/backup/backup_79/log_10_2
if=/home/oracle/dev/raw/log_10_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_78/lo_75
if=/home/oracle/dev/raw/lo_75 bs=1024k count=19001
dd of=/home/oracle/backup/backup_78/t_75
if=/home/oracle/dev/raw/t_75 bs=1024k count=12000
dd of=/home/oracle/backup/backup_78/undo_10
if=/home/oracle/dev/raw/undo_10 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_77/lo_76
if=/home/oracle/dev/raw/lo_76 bs=1024k count=19001
dd of=/home/oracle/backup/backup_77/t_76
if=/home/oracle/dev/raw/t_76 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_76/lo_77
if=/home/oracle/dev/raw/lo_77 bs=1024k count=19001
dd of=/home/oracle/backup/backup_76/t_77
if=/home/oracle/dev/raw/t_77 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_75/lo_78
if=/home/oracle/dev/raw/lo_78 bs=1024k count=19001
dd of=/home/oracle/backup/backup_75/t_78
if=/home/oracle/dev/raw/t_78 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_74/lo_79
if=/home/oracle/dev/raw/lo_79 bs=1024k count=19001
dd of=/home/oracle/backup/backup_74/t_79
if=/home/oracle/dev/raw/t_79 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_73/lo_80
if=/home/oracle/dev/raw/lo_80 bs=1024k count=19001
dd of=/home/oracle/backup/backup_73/t_80
if=/home/oracle/dev/raw/t_80 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_88/lo_81
if=/home/oracle/dev/raw/lo_81 bs=1024k count=19001
dd of=/home/oracle/backup/backup_88/t_81
if=/home/oracle/dev/raw/t_81 bs=1024k count=12000
dd of=/home/oracle/backup/backup_88/log_11_1
if=/home/oracle/dev/raw/log_11_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_87/lo_82
if=/home/oracle/dev/raw/lo_82 bs=1024k count=19001
dd of=/home/oracle/backup/backup_87/t_82
if=/home/oracle/dev/raw/t_82 bs=1024k count=12000
dd of=/home/oracle/backup/backup_87/log_11_2
if=/home/oracle/dev/raw/log_11_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_86/lo_83
if=/home/oracle/dev/raw/lo_83 bs=1024k count=19001
dd of=/home/oracle/backup/backup_86/t_83
if=/home/oracle/dev/raw/t_83 bs=1024k count=12000
```


© 2004 Hewlett Packard Company. All rights reserved

```
dd of=/home/oracle/backup/backup_86/undo_11
if=/home/oracle/dev/raw/undo_11 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_85/lo_84
if=/home/oracle/dev/raw/lo_84 bs=1024k count=19001
dd of=/home/oracle/backup/backup_85/t_84
if=/home/oracle/dev/raw/t_84 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_84/lo_85
if=/home/oracle/dev/raw/lo_85 bs=1024k count=19001
dd of=/home/oracle/backup/backup_84/t_85
if=/home/oracle/dev/raw/t_85 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_83/lo_86
if=/home/oracle/dev/raw/lo_86 bs=1024k count=19001
dd of=/home/oracle/backup/backup_83/t_86
if=/home/oracle/dev/raw/t_86 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_82/lo_87
if=/home/oracle/dev/raw/lo_87 bs=1024k count=19001
dd of=/home/oracle/backup/backup_82/t_87
if=/home/oracle/dev/raw/t_87 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_81/lo_88
if=/home/oracle/dev/raw/lo_88 bs=1024k count=19001
dd of=/home/oracle/backup/backup_81/t_88
if=/home/oracle/dev/raw/t_88 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_96/lo_89
if=/home/oracle/dev/raw/lo_89 bs=1024k count=19001
dd of=/home/oracle/backup/backup_96/t_89
if=/home/oracle/dev/raw/t_89 bs=1024k count=12000
dd of=/home/oracle/backup/backup_96/log_12_1
if=/home/oracle/dev/raw/log_12_1 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_95/lo_90
if=/home/oracle/dev/raw/lo_90 bs=1024k count=19001
dd of=/home/oracle/backup/backup_95/t_90
if=/home/oracle/dev/raw/t_90 bs=1024k count=12000
dd of=/home/oracle/backup/backup_95/log_12_2
if=/home/oracle/dev/raw/log_12_2 bs=1024k count=4100
} &

{
dd of=/home/oracle/backup/backup_94/lo_91
if=/home/oracle/dev/raw/lo_91 bs=1024k count=19001
dd of=/home/oracle/backup/backup_94/t_91
if=/home/oracle/dev/raw/t_91 bs=1024k count=12000
dd of=/home/oracle/backup/backup_94/undo_12
if=/home/oracle/dev/raw/undo_12 bs=1024k count=9000
} &

{
dd of=/home/oracle/backup/backup_93/lo_92
if=/home/oracle/dev/raw/lo_92 bs=1024k count=19001
dd of=/home/oracle/backup/backup_93/t_92
if=/home/oracle/dev/raw/t_92 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_92/lo_93
if=/home/oracle/dev/raw/lo_93 bs=1024k count=19001
dd of=/home/oracle/backup/backup_92/t_93
if=/home/oracle/dev/raw/t_93 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_91/lo_94
if=/home/oracle/dev/raw/lo_94 bs=1024k count=19001
dd of=/home/oracle/backup/backup_91/t_94
if=/home/oracle/dev/raw/t_94 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_90/lo_95
if=/home/oracle/dev/raw/lo_95 bs=1024k count=19001
dd of=/home/oracle/backup/backup_90/t_95
if=/home/oracle/dev/raw/t_95 bs=1024k count=12000
} &

{
dd of=/home/oracle/backup/backup_89/lo_96
if=/home/oracle/dev/raw/lo_96 bs=1024k count=19001
dd of=/home/oracle/backup/backup_89/t_96
if=/home/oracle/dev/raw/t_96 bs=1024k count=12000
} &
wait
date >> backup4.log
```


© 2004 Hewlett Packard Company. All rights reserved

Appendix C: ACID Scripts

```
_____________________________________
            a_query2.sql
_____________________________________
Rem
Rem $Header: aquery2.sql 07-aug-99.23:54:47 mpoess Exp $
Rem
Rem aquery2.sql
Rem
Rem  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
Rem
Rem    NAME
Rem      aquery2.sql - <one-line expansion of the name>
Rem
Rem  DESCRIPTION
Rem       Performs query on PARTSUPP for TPC-D benchmark
Rem       Isolation Test 5.
Rem       Asks user to input values for ps_partkey and ps_suppkey
Rem       The range for ps_partkey is 1 to 20000
Rem       The range for ps_suppkey is 1 to 1000
Rem       A valid combination is 46 and 47
Rem  Usage:   sqlplus tpcd/tpcd @a_query2 <ps_partkey>
<ps_suppkey>
Rem
Rem    MODIFIED   (MM/DD/YY)
Rem    mpoess      08/07/99 - Creation
Rem    mpoess      08/07/99 - Created
Rem
rem  DESCRIPTION
rem       Performs query on PARTSUPP for TPC-D benchmark
rem       Isolation Test 5.
rem       Asks user to input values for ps_partkey and ps_suppkey
rem       The range for ps_partkey is 1 to 20000
rem       The range for ps_suppkey is 1 to 1000
rem       A valid combination is 46 and 47

set serverout on;

select
'BEFORE PARTSUPP QUERY' as STAGE,
substr(TO_CHAR(sysdate,'YYYY-MM-DD HH:MI:SS'),1,20) as
CURRENT_TIME
from dual;

select *
from partsupp
where ps_partkey = &&1
and ps_suppkey = &&2;

select
'AFTER PARTSUPP QUERY' as STAGE,
substr(TO_CHAR(sysdate,'YYYY-MM-DD HH:MI:SS'),1,20) as
CURRENT_TIME
from dual;

exit;


_____________________________________
            a_query.sql
_____________________________________
Rem
Rem $Header: a_query.sql 06-aug-99.10:51:10 mpoess Exp $
Rem
Rem a_query.sql
Rem
Rem  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
Rem
Rem    NAME
Rem      a_query.sql - <one-line expansion of the name>
Rem
rem  DESCRIPTION
Rem       Performs ACID Query for TPC-D benchmark.
Rem       Asks user to input values for o_key
Rem       The range of okey is 1 to 600000
Rem
=================================================
===============
Rem
Rem Usage:   sqlplus tpcd/tpcd @a_query <o_key>
Rem
Rem
Rem    MODIFIED   (MM/DD/YY)
Rem    mpoess      08/06/99 - Creation
Rem    mpoess      08/06/99 - Created
Rem

set serverout on;

select
'BEFORE ACID QUERY' as STAGE,
substr(TO_CHAR(sysdate,'YYYY-MM-DD HH:MI:SS'),1,20) as
CURRENT_TIME
from dual;

select SUM(trunc(trunc(l_extendedprice * (1-l_discount),2) *
(1+l_tax),2)) AS RESULT
from lineitem
where l_orderkey = &&1;

select
'AFTER ACID QUERY' as STAGE,
substr(TO_CHAR(sysdate,'YYYY-MM-DD HH:MI:SS'),1,20) as
CURRENT_TIME
from dual;

exit;


_____________________________________
            atom.sh
_____________________________________
#!/bin/ksh
#
# $Header: atom.sh 08-aug-99.13:48:02 mpoess Exp $
#
# atom.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#    NAME
#      atom.sh - <one-line expansion of the name>
#
#  DESCRIPTION
#      Performs atomicity tests.
#      Usage: atom.sh [-n iter] [-p prog] [-u usr/pswd] -h
#
```


© 2004 Hewlett Packard Company. All rights reserved

```
#     Options: See usage below
#
#   NOTES
#     <other useful comments, qualifications, etc.>
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/08/99 - Creation
#   mpoess      08/08/99 - Creation
#

. $KIT_DIR/env

OH=$ORACLE_HOME
# ACID_DIR=$TPCD_KIT_DIR/audit set in env
OUT_DIR=$ACID_OUT
DURA_DIR=$ACID_DIR/dura

usage() {

  echo ""
  echo "Usage: $0 [-n iter] [-p prog] [-u usr/pswd] -h"
  echo ""
  echo "-n iter     : number of iterations, default is 100"
  echo "-p prog     : program to run, default is atranspl.ott"
  echo "-u usr/pswd : user/password combo for database access, default
is tpcd/tpcd"
  echo "-h          : print this usage summary"
  exit 1;
}

ITER=3
SF=1
PROG=$KIT_DIR/utils/atranspl
OUT=${OUT_DIR}/atom
USER=${DATABASE_USER}

set -- `getopt "n:p:u:h" "$@"` || usage

while :
do
  case "$1" in
  -n) shift; ITER=$1;;
  -p) shift; PROG=$1;;
  -u) shift; USER=$1;;
  -h) usage; exit 0;;
  --) break;;
  esac
  shift
done

echo "Starting Atomicity Test at `date`..."
echo ""
echo "Performing $ITER ACID transactions with COMMIT"
echo ""

$KIT_DIR/utils/randkey $ITER $SF u$USER | $PROG 1 1 1 0 u$USER
> ${OUT}c 2>&1

echo "ACID transactions with COMMIT ended. Output in ${OUT}c"
echo ""
echo "Performing $ITER ACID transactions with ROLLBACK"
echo ""

$KIT_DIR/utils/randkey $ITER $SF u$USER | $PROG 1 1 0 0 u$USER
> ${OUT}r 2>&1

echo "ACID transactions with ROLLBACK ended. Output in ${OUT}r"
echo ""
echo "Ending Atomicity Test at `date`..."
```

```
_______________________________________
          atranspl.c
_______________________________________
/* Copyright (c) 2001, 2002, Oracle Corporation.  All rights reserved.  */

/*

  NAME
    atranspl.c - <one-line expansion of the name>

  DESCRIPTION
    TPC-HR benchmark ACID transaction driver, OCI version 8

  NOTES
    <other useful comments, qualifications, etc.>

  MODIFIED   (MM/DD/YY)
  mpoess      10/23/02 - mpoess_update_from_visa
  mpoess      10/17/01 - add parameter in ACIDinit
  mpoess      02/22/01 - enlarge timing array
  mpoess      01/04/01 - Creation

*/

#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>

#include "atranspl.h"

/* Declare error handling functions */

double gettime();
void sql_error();
void usage();
void ACIDinit();
void ACIDexit();
int atoi();
void srand48();
long lrand48();

/* declarations for ORDERS */

int o_key = 0;
double o_tprice = 0.0;
double o_newtprice = 0.0;

/* declarations for LINEITEM */

int l_key = 0;
int l_pkey = 0;
int l_skey = 0;

int l_quan = 0;
int l_newquan = 0;
double l_eprice = 0.0;
double l_neweprice = 0.0;
double l_disc = 0.0;
double l_tax = 0.0;

sb2 l_npricei;

/* other declarations */

int delta = 0;
double rprice;
double cost;
```


© 2004 Hewlett Packard Company. All rights reserved

```
int proc_no = 1;          /* process number, global              */
int num_streams = 1;      /* number of transaction streams       */
int trig = 0;             /* Trigger Time                        */
int slp = 0;              /* Sleep Time                          */

int logfile;              /* fdes for logfile for durability (optional)   */
int outfile = 1;          /* output file (optional)              */
#ifdef LINUX
  FILE *infile;     /* input file (optional)              */
#else
  FILE *infile = stdin;     /* input file (optional)              */
                        /* in the format of <o_key> <delta>    */
#endif
char lname[UNAME_LEN];    /* username/passwd combo              */
char *passwd;             /* pointer to password                */

char buf[WRITE_BUF_LEN];  /* buffer to write                    */

unsigned flag = (unsigned) 0;       /* flag to store all sorts of options  */

#define INFILE  0x01u
#define OUTFILE 0x02u
#define LOGFILE 0x04u
#define COMMIT  0x08u
#define DELTA   0x10u

double tr_end = 0.0;      /* transaction end time               */
double tr_start = 0.0;    /* transaction start time             */

int num_iter = 0;         /* number of iterations               */

time_t curr_time;         /* Current Time                       */

/* OCI handles */

OCIEnv *tpcenv = NULL;
OCIServer *tpcsrv = NULL;
OCIError *errhp = NULL;
OCISvcCtx *tpcsvc = NULL;
OCISession *tpcusr = NULL;
OCIStmt *curi = NULL;
OCIStmt *curr = NULL;
OCIStmt *cure1 = NULL;
OCIStmt *cure2 = NULL;

/* OCI bind handles */

#ifdef NOLKEY
OCIBind *l_keyi_bp = NULL;
OCIBind *o_keyi_bp = NULL;
#endif /* NOLKEY */

OCIBind *l_key_bp = NULL;
OCIBind *o_key_bp = NULL;
OCIBind *delta_bp = NULL;
OCIBind *l_pkey_bp = NULL;
OCIBind *l_skey_bp = NULL;
OCIBind *l_quan_bp = NULL;
OCIBind *l_newquan_bp = NULL;
OCIBind *l_tax_bp = NULL;
OCIBind *l_disc_bp = NULL;
OCIBind *l_eprice_bp = NULL;
OCIBind *l_neweprice_bp = NULL;
OCIBind *o_tprice_bp = NULL;
OCIBind *o_newtprice_bp = NULL;
OCIBind *rprice_bp = NULL;
OCIBind *cost_bp = NULL;

OCIBind *l_neweprice1_bp = NULL;
OCIBind *l_newquan1_bp = NULL;
OCIBind *o_key1_bp = NULL;
OCIBind *l_key1_bp = NULL;

OCIBind *o_newtprice2_bp = NULL;
OCIBind *o_key2_bp = NULL;

sword status = OCI_SUCCESS;  /* OCI return value */

char sqlstmt[1024];

/* usage: prints the usage of the program */

void usage()
{

  fprintf (stderr,"\nUsage: atrans.o[st]t <proc_no> <num_streams>
<commit> <delta>\n[i<pathname for input>] [o<pathname for output>]
[d<pathname for durability file>] [u<uid/passwd>] \n\n");

  fprintf(stderr,"    proc_no      :the process number within this ACID\n");
  fprintf(stderr,"    num_streams  :the total number of ACID transaction
streams\n");
  fprintf(stderr,"    commit       :1 to commit transaction, abort
otherwise\n\n");
  fprintf(stderr,"    delta        :1 to generate new random delta, otherwise
obtain delta from input\n\n");
  fprintf(stderr,"    OPTIONAL PARAMETERS:\n");
  fprintf(stderr,"    i<pathname for input>      :full path name for input file
- default is stdin\n");
  fprintf(stderr,"    o<pathname for output>     :full path name for output
file - default is stdout\n");
  fprintf(stderr,"    d<pathname for durability> :full path name for
durability success file - must specify for durability test\n");
  fprintf(stderr,"    u<uid/passwd>              :Username/Password string -
default is tcpd/tpcd\n");
  fprintf(stderr,"    t<trigger>                 :Trigger Time - sleep <trigger>
seconds before start\n\n");
  fprintf(stderr,"    s<sleep>                   :Sleep Time - sleep <sleep>
seconds before commit or rollback\n\n");
  exit(-1);

}


void ACIDexit() {

  OCILogoff(tpcsvc,errhp);
  OCIhfree(tpcenv,OCI_HTYPE_STMT);
  OCIhfree(tpcsvc,OCI_HTYPE_SVCCTX);
  OCIhfree(tpcsrv,OCI_HTYPE_SERVER);
  OCIhfree(tpcusr,OCI_HTYPE_SESSION);

}



/* type: 0 if environment handle is passed, 1 if error handle is passwd */

void sql_error(errhp,status,type)
    OCIError *errhp;
    sword status;
    sword type;
{
  char msg[2048];
  ub4 errcode;
  ub4 msglen;
  int i,j;

  switch(status) {
```


© 2004 Hewlett Packard Company. All rights reserved

```
  case OCI_SUCCESS_WITH_INFO:
    fprintf(stderr, "Error: Statement returned with info.\n");
    if (type)
      (void) OCIErrorGet(errhp,1,NULL,(sb4*) &errcode, (text*) msg,
                    2048, OCI_HTYPE_ERROR);
    else
      (void) OCIErrorGet(errhp,1,NULL,(sb4*) &errcode, (text*) msg,
                    2048, OCI_HTYPE_ENV);
    fprintf(stderr,"%s\n",msg);
    break;
  case OCI_ERROR:
    fprintf(stderr, "Error: OCI call error.\n");
    if (type)
      (void) OCIErrorGet(errhp,1,NULL, (sb4 *) &errcode, (text*) msg,
                    2048,OCI_HTYPE_ERROR);
    else
      (void) OCIErrorGet(errhp,1,NULL, (sb4 *) &errcode, (text*) msg,
                    2048,OCI_HTYPE_ENV);
    fprintf(stderr,"%s\n",msg);
    break;
  case OCI_INVALID_HANDLE:
    fprintf(stderr, "Error: Invalid Handle.\n");
    if (type)
      (void) OCIErrorGet(errhp,1,NULL, (sb4 *) &errcode, (text*) msg,
                    2048,OCI_HTYPE_ERROR);
    else
      (void) OCIErrorGet(errhp,1,NULL, (sb4 *) &errcode, (text*) msg,
                    2048,OCI_HTYPE_ENV);
    fprintf(stderr,"%s\n",msg);
    break;
  }
  /* Rollback just in case */

  (void) OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);

  fprintf(stderr, "Exiting Oracle...\n");
  fflush(stderr);

  ACIDexit();

  exit(1);
}

#ifdef LINUX
int main(argc,argv)
#else
void main(argc,argv)
#endif
     int argc;
     char *argv[];
{

  int i;
  char line[64];
  ub4 errcode;
  char msg[2048];
  int need_commit = 0;

  /* Initialize some variables */
#ifdef LINUX
  infile=fopen("/dev/stdin","r");
#endif
  strcpy((char *) lname, "tpcd/tpcd");

  if ((argc > 10) || (argc < 5)) {
    usage();
  }

  /* argv[1] -- Process Number */

  proc_no = atoi(argv[1]);

  /* argv[2] -- Number of Streams */

  num_streams = atoi(argv[2]);

  /* argv[3] -- Commit? */

  if (atoi(argv[3]) == 1)
    BIS(flag, COMMIT);

  /* argv[4] -- Delta? */

  if (atoi(argv[4]) == 1)
    BIS(flag, DELTA);

  /* Process  optional parameters */

  argc -= 4;
  argv += 4;

  while(--argc) {
    ++argv;
    switch(argv[0][0]) {
    case 'u':
      strncpy((char *) lname, ++(argv[0]), UNAME_LEN);
      if (strchr((char *) lname, '/') == NULL) {
        fprintf(stderr, "Login name must be in the format of
userid/passwd\n");
        usage();
        exit(-1);
      }
      break;
    case 'i':
      if ((infile = fopen(++(argv[0]), "r")) == NULL) {
        fprintf(stderr,"Cannot open input file %s\n", argv[0]);
        fprintf(stderr,"%s\n",strerror(errno));
        exit(-1);
      }
      BIS(flag, INFILE);
      break;
    case 'o':
      if ((outfile = open(++(argv[0]), (O_RDWR | O_SYNC | O_CREAT),
S_IRWXU)) == -1) {
        fprintf(stderr,"Cannot open output file %s\n", argv[0]);
        fprintf(stderr,"%s\n",strerror(errno));
        exit(-1);
      }
      BIS(flag, OUTFILE);
      break;
    case 'd':
      if ((logfile = open(++(argv[0]), (O_RDWR | O_SYNC | O_CREAT),
S_IRWXU)) == -1) {
        fprintf(stderr,"Cannot open durability success file %s\n", argv[0]);
        fprintf(stderr,"%s\n",strerror(errno));
        exit(-1);
      }
      BIS(flag, LOGFILE);
      break;
    case 'b':
      num_iter = atoi(++(argv[0]));
      break;
    case 't':
      trig = atoi(++(argv[0]));
      break;
    case 's':
      slp = atoi(++(argv[0]));
      break;
    default:
      fprintf(stderr, "Unknown argument %s\n", argv[0]);
```


© 2004 Hewlett Packard Company. All rights reserved

```
      usage();
      break;
    }
  }

  FPRTF(outfile,"----------------------------------------------\n");

  /* Initialize the cursors etc. */

  (void) ACIDinit();

  /* sleep for some time (triggering) */

  sleep(trig);

  /* start doing the ACID transactions */

  tr_start = gettime();

  /* The number of iteration we will run depends on the number of */
  /* input lines                                                  */

  while (fgets(line, 64, infile) != NULL) {

#ifdef NOLKEY
    sscanf(line, "%d %d\n", &o_key, &delta);

    /* Obtain l_key from l_key query */

    OCIsexec(tpcsvc,curi,errhp,1);

    /* l_key is the highest l_linenumber available.  We need to pick */
    /* at random a number between 1..l_key.                          */

    l_key = (int) ((lrand48() % l_key) + 1);
#else
    sscanf(line, "%d %d %d\n", &o_key, &l_key, &delta);
#endif /* NOLKEY */

    /* Generate delta if necessary */

    if (BIT(flag, DELTA))
      delta = (int) (floor((drand48() * 100)) + 1);

    /* Now, we are ready to run the ACID transaction. */

    curr_time = time(NULL);

    FPRTF2(outfile, "Starting ACID transaction %d at %s...\n",
(++num_iter),
           ctime(&curr_time));

    FPRTF1(outfile, "o_key: %d\n", (int) o_key);
    FPRTF1(outfile, "l_key: %d\n", (int) l_key);
    FPRTF1(outfile, "delta: %d\n", (int) delta);

    OCIsexec(tpcsvc,curr,errhp,1);

    curr_time = time(NULL);

    if (!BIT(flag, LOGFILE)) {
    FPRTF1(outfile, "BEFORE COMMIT/ROLLBACK
TRANSACTION at %s\n", ctime(&curr_time));
    FPRTF1(outfile, "l_extendedprice: %.2f\n", l_eprice);
    FPRTF1(outfile, "l_quantity:      %d\n", (int) l_quan);
    FPRTF1(outfile, "o_totalprice:    %.2f\n\n", o_tprice);
    }

    FPRTF1(outfile, "Sleep %d seconds before
COMMIT/ROLLBACK...\n\n", slp);
    sleep(slp);

    /* Shall we commit? */

    if (BIT(flag, COMMIT)) {
      need_commit = 1;
      while (need_commit) {
        if((status=OCITransCommit(tpcsvc,errhp,OCI_DEFAULT)) !=
OCI_SUCCESS) {
            OCIrol(tpcsvc,errhp);
            OCIsexec(tpcsvc,curr,errhp,1);
        } else {
           need_commit = 0;
           curr_time = time(NULL);
           FPRTF2(outfile, "ACID Transaction iteration %d COMMITED
at %s\n",
             num_iter, ctime(&curr_time));
        }
      }
    } else {
      OCIrol(tpcsvc,errhp);
      curr_time = time(NULL);
      FPRTF2(outfile, "ACID Transaction iteration %d ROLLBACK at
%s\n",
           num_iter, ctime(&curr_time));
    }

    /* Report all results to outfile and if necessary, to success file. */

    /* Report initial and new values for o_totalprice, l_extendedprice, */
    /* l_quantity.                                                      */

    /*
    curr_time = time(NULL);
    FPRTF1(outfile, "Transaction Completed at %s\n",
ctime(&curr_time));
    */

    /* Get the values in LINEITEM and ORDERS after the transaction */

    if (BIT(flag, LOGFILE)) {
      FPRTF1(logfile, "p_key:      %d\n", (int) l_pkey);
      FPRTF1(logfile, "s_key:      %d\n", (int) l_skey);
      FPRTF1(logfile, "o_key:      %d\n", (int) o_key);
      FPRTF1(logfile, "l_key:      %d\n", (int) l_key);
      FPRTF1(logfile, "delta:      %d\n", (int) delta);
      FPRTF1(logfile, "Transaction Completed at %s\n",
ctime(&curr_time));
      FPRTF(logfile, "----------------------------------------------\n");
    } else {

      OCIsexec(tpcsvc,cure1,errhp,1);
      OCIsexec(tpcsvc,cure2,errhp,1);

      FPRTF(outfile, "AFTER TRANSACTION:\n");
      FPRTF1(outfile, "l_extendedprice: %.2lf\n", l_neweprice);
      FPRTF1(outfile, "l_quantity:      %d\n", (int) l_newquan);
      FPRTF1(outfile, "o_totalprice:    %.2lf\n\n", o_newtprice);
      FPRTF1(outfile, "l_tax:           %.2lf\n", l_tax);
      FPRTF1(outfile, "l_discount:      %.2lf\n", l_disc);
      FPRTF1(outfile, "rprice:          %.2lf\n", rprice);
      FPRTF1(outfile, "cost:            %.2lf\n", cost);
      FPRTF(outfile, "----------------------------------------------\n");
    }
  }

  tr_end = gettime();

  if (!BIT(flag,LOGFILE)) {
    FPRTF1(outfile, "Start Time: %.2f\n", tr_start);
```


© 2004 Hewlett Packard Company. All rights reserved

```
   FPRTF1(outfile, "End Time: %.2f\n", tr_end);
   FPRTF1(outfile, "Elapsed Time: %.2f\n", (tr_end - tr_start));
   FPRTF1(outfile, "Transaction Count: %d\n", num_iter);
   FPRTF1(outfile, "Transaction Rate: %.2f\n", num_iter/(tr_end -
tr_start));
  } else {
   FPRTF1(logfile, "Start Time: %.2f\n", tr_start);
   FPRTF1(logfile, "End Time: %.2f\n", tr_end);
   FPRTF1(logfile, "Elapsed Time: %.2f\n", (tr_end - tr_start));
   FPRTF1(logfile, "Transaction Count: %d\n", num_iter);
  }

  /*  Disconnect from ORACLE. */

  if (BIT(flag, INFILE))
   fclose(infile);
  if (BIT(flag, OUTFILE))
   close(outfile);
  if (BIT(flag, LOGFILE))
   close(logfile);

  ACIDexit();

  exit(0);
}


void ACIDinit()
{

  /* run random seed */

  srand48(getpid());

  /* Connect to ORACLE.  Program will call sql_error()
     if an error occurs in connecting to the default database. */

  (void) OCIInitialize(OCI_DEFAULT,(dvoid *)0,0,0,0);
  if((status=OCIEnvInit((OCIEnv **)&tpcenv,OCI_DEFAULT,0,(dvoid
**)0)) !=
     OCI_SUCCESS)
   sql_error(tpcenv, status, 0);

  OCIhalloc(tpcenv,&errhp,OCI_HTYPE_ERROR);
  OCIhalloc(tpcenv,&curi,OCI_HTYPE_STMT);
  OCIhalloc(tpcenv,&curr,OCI_HTYPE_STMT);
  OCIhalloc(tpcenv,&cure1,OCI_HTYPE_STMT);
  OCIhalloc(tpcenv,&cure2,OCI_HTYPE_STMT);
  OCIhalloc(tpcenv,&tpcsvc,OCI_HTYPE_SVCCTX);
  OCIhalloc(tpcenv,&tpcsrv,OCI_HTYPE_SERVER);
  OCIhalloc(tpcenv,&tpcusr,OCI_HTYPE_SESSION);

  /* Disables auto commit */
/*
  if (ocof(&tpclda)) {
   sql_error(&tpclda, &tpclda);
   ologof(&tpclda);
   exit(-1);
  }
*/

  /* get username and password */

  passwd = strchr(lname, '/');
  *passwd = '\0';
  passwd++;

  if ((status = OCIServerAttach(tpcsrv,errhp,(text
*)0,0,OCI_DEFAULT)) != OCI_SUCCESS)
   sql_error(errhp,status,1);


OCIaset(tpcsvc,OCI_HTYPE_SVCCTX,tpcsrv,0,OCI_ATTR_SERVER
,errhp);

OCIaset(tpcusr,OCI_HTYPE_SESSION,lname,strlen(lname),OCI_ATT
R_USERNAME,
        errhp);

OCIaset(tpcusr,OCI_HTYPE_SESSION,passwd,strlen(passwd),OCI_A
TTR_PASSWORD,
        errhp);

  if ((status = OCISessionBegin(tpcsvc, errhp, tpcusr,
OCI_CRED_RDBMS,
                        OCI_DEFAULT)) != OCI_SUCCESS)
    sql_error(errhp,status,1);


OCIaset(tpcsvc,OCI_HTYPE_SVCCTX,tpcusr,0,OCI_ATTR_SESSIO
N,errhp);

  /* Enable session parallel dml */

  sprintf((char *) sqlstmt, PDMLTXT);
  OCIStmtPrepare(curi,errhp,(text *)sqlstmt,strlen((char *)sqlstmt),
            OCI_NTV_SYNTAX,OCI_DEFAULT);
  OCIsexec(tpcsvc,curi,errhp,1);

  /* Enable session parallel ddl */

  /*sprintf((char *) sqlstmt, PDDLTXT);
  OCIStmtPrepare(curi,errhp,(text *)sqlstmt,strlen((char *)sqlstmt),
            OCI_NTV_SYNTAX,OCI_DEFAULT);
  OCIsexec(tpcsvc,curi,errhp,1);*/


  /* Make session serializable */

  sprintf ((char *) sqlstmt, ISOTXT);
  OCIStmtPrepare(curi,errhp,(text *)sqlstmt,strlen((char *)sqlstmt),
            OCI_NTV_SYNTAX,OCI_DEFAULT);
  OCIsexec(tpcsvc,curi,errhp,1);

  /* Set optimizer_index_cost_adj = 25 */

  sprintf ((char *) sqlstmt, OICATXT);
  OCIStmtPrepare(curi,errhp,(text *)sqlstmt,strlen((char *)sqlstmt),
            OCI_NTV_SYNTAX,OCI_DEFAULT);
  OCIsexec(tpcsvc,curi,errhp,1);

  curr_time = time(NULL);
  printf("\nConnected to ORACLE as user: %s at %s\n\n", lname,
ctime(&curr_time));


#ifdef NOLKEY
  /* Open and Parse cursor for query to choose determine l_key. */
  /* Binds l_key to :l_key.                                     */

  sprintf((char *) sqlstmt,SQLTXT1);
  OCIStmtPrepare(curi,errhp,sqlstmt,strlen((char
*)sqlstmt),OCI_NTV_SYNTAX,OCI_DEFAULT);



OCIbbname(curi,&l_keyi_bp,errhp,":l_key",ADR(l_key),SIZ(l_key),SQ
LT_INT);
```


© 2004 Hewlett Packard Company. All rights reserved

```
OCIbbname(curi,&o_keyi_bp,errhp,":o_key",ADR(o_key),SIZ(o_key),
SQLT_INT);

#endif /* NOLKEY */

  /* Open and Parse cursor for the ACID transaction.          */

  sprintf((char *) sqlstmt,SQLTXT2);
  OCIStmtPrepare(curr,errhp,(text *)sqlstmt,strlen((char *)sqlstmt),
            OCI_NTV_SYNTAX,OCI_DEFAULT);

  /* bind variables */


OCIbbname(curr,l_key_bp,errhp,":l_key",ADR(l_key),SIZ(l_key),SQL
T_INT);

OCIbbname(curr,o_key_bp,errhp,":o_key",ADR(o_key),SIZ(o_key),SQ
LT_INT);

OCIbbname(curr,delta_bp,errhp,":delta",ADR(delta),SIZ(delta),SQLT_I
NT);

OCIbbname(curr,l_pkey_bp,errhp,":l_pkey",ADR(l_pkey),SIZ(l_pkey),
SQLT_INT);

OCIbbname(curr,l_skey_bp,errhp,":l_skey",ADR(l_skey),SIZ(l_skey),S
QLT_INT);

OCIbbname(curr,l_quan_bp,errhp,":l_quan",ADR(l_quan),SIZ(l_quan),
SQLT_INT);
  OCIbbname(curr,l_newquan_bp,errhp,":l_newquan",ADR(l_newquan),
        SIZ(l_newquan),SQLT_INT);

OCIbbname(curr,l_tax_bp,errhp,":l_tax",ADR(l_tax),SIZ(l_tax),SQLT_
FLT);

OCIbbname(curr,l_disc_bp,errhp,":l_disc",ADR(l_disc),SIZ(l_disc),SQ
LT_FLT);

OCIbbname(curr,l_eprice_bp,errhp,":l_eprice",ADR(l_eprice),SIZ(l_epr
ice),
        SQLT_FLT);

OCIbbname(curr,l_neweprice_bp,errhp,":l_neweprice",ADR(l_newepric
e),
        SIZ(l_neweprice),SQLT_FLT);


OCIbbname(curr,o_tprice_bp,errhp,":o_tprice",ADR(o_tprice),SIZ(o_tp
rice),
        SQLT_FLT);

OCIbbname(curr,o_newtprice_bp,errhp,":o_newtprice",ADR(o_newtpri
ce),
        SIZ(o_newtprice), SQLT_FLT);
  OCIbbname(curr,rprice_bp,errhp,":rprice",ADR(rprice),SIZ(rprice),
SQLT_FLT);
  OCIbbname(curr,cost_bp,errhp,":cost",ADR(cost),SIZ(cost),
SQLT_FLT);

  /* Open & Parse cursor for end values query */

  sprintf((char *) sqlstmt,SQLTXT3);
  OCIStmtPrepare(cure1,errhp,(text *)sqlstmt,strlen((char *)sqlstmt),
            OCI_NTV_SYNTAX,OCI_DEFAULT);

  sprintf((char *) sqlstmt,SQLTXT4);
  OCIStmtPrepare(cure2,errhp,(text *)sqlstmt,strlen((char *)sqlstmt),
            OCI_NTV_SYNTAX,OCI_DEFAULT);

  /* bind variables */


OCIbbname(cure1,l_neweprice1_bp,errhp,":l_neweprice",ADR(l_newep
rice),
        SIZ(l_neweprice),SQLT_FLT);

OCIbbname(cure1,l_newquan1_bp,errhp,":l_newquan",ADR(l_newqua
n),
        SIZ(l_newquan),SQLT_INT);

OCIbbname(cure1,o_key1_bp,errhp,":o_key",ADR(o_key),SIZ(o_key),
SQLT_INT);

OCIbbname(cure1,l_key1_bp,errhp,":l_key",ADR(l_key),SIZ(l_key),S
QLT_INT);


OCIbbname(cure2,o_newtprice2_bp,errhp,":o_newtprice",ADR(o_newt
price),
        SIZ(o_newtprice),SQLT_FLT);

OCIbbname(cure2,o_key2_bp,errhp,":o_key",ADR(o_key),SIZ(o_key),
SQLT_INT);

}
```

atranspl.h

```
/* Copyright (c) 2001, 2002, Oracle Corporation.  All rights reserved.  */

/*
  NAME
    atranspl.h - <one-line expansion of the name>

  DESCRIPTION

  MODIFIED   (MM/DD/YY)
  mpoess     10/23/02 - mpoess_update_from_visa
  mpoess     10/17/01 - add TXT parameter
  mpoess     04/09/01 - add hint to find max linenumber
  mpoess     01/04/01 - Creation

*/
#ifndef ATRANSPL_H

#define ATRANSPL_H

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/param.h>
#include <sys/types.h>
#include <time.h>
#include <errno.h>
#include <math.h>

#include <oratypes.h>
#ifndef OCIDFN
#include <ocidfn.h>
#endif /* OCIDFN */

#ifndef OCI_ORACLE
#include <oci.h>
#endif /* OCI_ORACLE */
```


© 2004 Hewlett Packard Company. All rights reserved

```
/*
#ifdef __STDC__
#include <ociapr.h>
#else
#include <ocikpr.h>
#endif */ /* __STDC__ */

extern int errno;

#ifndef NULL
#define NULL 0
#endif

#ifndef NULLP
# define NULLP (void *)NULL
#endif /* NULLP */

#ifndef DISCARD
# define DISCARD (void)
#endif

#ifndef sword
# define sword int
#endif

#ifndef ub1
#define ub1 unsigned char
#endif

#define UNAME_LEN 64
#define WRITE_BUF_LEN 1024

#define NA              -1      /* ANSI SQL NULL */
#define VER7            2
#define NOT_SERIALIZABLE  8177  /* ORA-08177: transaction not
serializable */
#define WRITE_BUF_LEN 1024

#define ADR(object) ((ub1 *)&(object))
#define SIZ(object) ((sword)sizeof(object))
#define BIS(flg,mask) (unsigned) (flg |= (unsigned) mask)
#define BIT(flg,mask) (unsigned) ((unsigned) flg & (unsigned) mask)

#define FPRTF(fd,s) \
{sprintf(buf,s); write(fd, buf, strlen(s));}
#define FPRTF1(fd,s,p) \
{sprintf(buf,s,p); write(fd, buf, strlen(buf));}
#define FPRTF2(fd,s,p1,p2) \
{sprintf(buf,s,p1,p2); write(fd, buf, strlen(buf));}

#define OCIhalloc(envh,hndl,htyp) \
   if((status=OCIHandleAlloc((dvoid *)envh,(dvoid
**)hndl,htyp,0,(dvoid **)0))!=OCI_SUCCESS) \
      sql_error(envh,status,0); \
   else \
      DISCARD 0

#define OCIhfree(hndl,htyp) \
   if((status=OCIHandleFree((dvoid *)hndl,htyp)) == OCI_SUCCESS) \
      fprintf(stderr, "Error freeing handle of type %d\n", htyp)

#define OCIaget(hndl,htyp,attp,size,atyp,errh) \
   if((status=OCIAttrGet((dvoid *)hndl,htyp,(dvoid *)attp,(dvoid
*)size,atyp,errh)) != OCI_SUCCESS) \
      sql_error(errh,status,1); \
   else \
      DISCARD 0

#define OCIaset(hndl,htyp,attp,size,atyp,errh) \
   if((status=OCIAttrSet((dvoid *)hndl,htyp,(dvoid
*)attp,size,atyp,errh)) != OCI_SUCCESS) \
      sql_error(errh,status,1); \
   else \
      DISCARD 0

#define OCIsexec(svch,stmh,errh,iter) \

if((status=OCIStmtExecute(svch,stmh,errh,iter,0,NULL,NULL,OCI_DE
FAULT)) != OCI_SUCCESS) \
      sql_error(errh,status,1); \
   else \
      DISCARD 0

#define OCIbbname(stmh,bindp,errh,sqlvar,progv,progvl,ftype) \
   if((status=OCIBindByName(stmh,&bindp,errh,(text
*)sqlvar,strlen(sqlvar), \
            progv,progvl,ftype,0,0,0,0,0,OCI_DEFAULT)) !=
OCI_SUCCESS) \
      sql_error(errh,status,1); \
   else \
      DISCARD 0

#define OCIbbnamei(stmh,bindp,errh,sqlvar,progv,progvl,ftype,indp) \
   if((status=OCIHandleAlloc((dvoid *)stmh,(dvoid
**)&bindp,OCI_HTYPE_BIND, \
                     0,(dvoid **)0))!=OCI_SUCCESS) \
      sql_error(stmh,status,0); \
   if((status=OCIBindByName(stmh,&bindp,errh,(text
*)sqlvar,strlen(sqlvar), \
            progv,progvl,ftype,indp,0,0,0,0,OCI_DEFAULT)) !=
OCI_SUCCESS) \
      sql_error(errh,status,1); \
   else \
      DISCARD 0

#define OCIcom(svcp,errh) \
    if((status=OCITransCommit(svcp,errh,OCI_DEFAULT)) !=
OCI_SUCCESS) \
       sql_error(errh,status,1); \
    else \
       DISCARD 0

#define OCIrol(svcp,errh) \
    if((status=OCITransRollback(svcp,errh,OCI_DEFAULT)) !=
OCI_SUCCESS) \
       sql_error(errh,status,1); \
    else \
       DISCARD 0

#define ISOTXT "alter session set isolation_level = serializable"
#define PDMLTXT "alter session force parallel dml parallel (degree 2)"
#define PDDLTXT "alter session force parallel ddl parallel (degree 2)"
#define OICATXT "alter session set optimizer_index_cost_adj=25"

#define SQLTXT1 "BEGIN SELECT /*+ index(lineitem,i_l_orderkey)
*/ MAX(l_linenumber) INTO :l_key FROM lineitem \
WHERE l_orderkey = :o_key; END;"

#define SQLTXT2 "BEGIN d_atrans.doatrans(:l_key, :o_key, :delta,
:l_pkey, \
:l_skey, :l_quan, :l_newquan, :l_tax, :l_disc, :l_eprice, :l_neweprice, \
:o_tprice, :o_newtprice, :rprice, :cost); END;"

#define SQLTXT3 "BEGIN SELECT l_extendedprice, l_quantity \
INTO :l_neweprice, :l_newquan \
FROM lineitem \
WHERE l_orderkey = :o_key \
AND l_linenumber = :l_key; END;"
```


© 2004 Hewlett Packard Company. All rights reserved

```
#define SQLTXT4 "BEGIN SELECT o_totalprice INTO :o_newtprice \
FROM orders \
WHERE o_orderkey = :o_key; END;"

#define SQLTXT5 "BEGIN SELECT l_extendedprice, l_quantity \
INTO :l_eprice, :l_quan \
FROM lineitem \
WHERE l_orderkey = :o_key \
AND l_linenumber = :l_key; END;"

#define SQLTXT6 "BEGIN SELECT o_totalprice INTO :o_tprice \
FROM orders \
WHERE o_orderkey = :o_key; END;"

#endif /* ATRANSPL_H */
```

---

ckpt.sh

```
#!/bin/ksh
#
# $Header: ckpt.sh 08-aug-99.17:37:07 mpoess Exp $
#
# ckpt.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     ckpt.sh - <one-line expansion of the name>
#
#  DESCRIPTION
#     Usage: ckpt.sh
#     Start database checkpoint
#
#   NOTES
#     <other useful comments, qualifications, etc.>
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/08/99 - Creation
#   mpoess      08/08/99 - Creation
#

. $KIT_DIR/env

sqlplus -s /NOLOG << !

        connect / as sysdba;
        alter system switch logfile;
        alter system switch logfile;
        exit;
!
```

---

cnt_hist.sql

```
select count(*) from history;
exit;
```

---

consist.sh

```
#!/bin/ksh
#
# $Header: consist.sh 08-aug-99.14:20:51 mpoess Exp $
#
# consist.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     consist.sh - <one-line expansion of the name>
#
#  DESCRIPTION
#     Performs consistency tests.
#     Usage: consist.sh [-n iter] [-s number of stream] [-p prog]
#                [-u usr/pswd] -h
#
#     Options: See usage below
#
#   NOTES
#     <other useful comments, qualifications, etc.>
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/08/99 - Creation
#   mpoess      08/08/99 - Creation
#

. $KIT_DIR/env

OH=$ORACLE_HOME
# ACID_DIR=$TPCD_KIT_DIR/audit set in env
OUT_DIR=$ACID_OUT

KEY=$OUT_DIR/key$$_
OUTFILE=${OUT_DIR}/consrte
CON1=${OUT_DIR}/conb
CON2=${OUT_DIR}/cona
CHK=${OUT_DIR}/consckpt

/bin/rm -rf ${KEY}* $CON1 $CON2 $OUTFILE $CHK

trap "/bin/rm -rf ${KEY}*; exit 1" 1 2 3 15

STREAM=${NUM_STREAMS}
let STREAM="$STREAM + 1" # add one for the update stream
ITER=100
PROG=atranspl
USER=${DATABASE_USER}
CK=10

usage() {

  echo ""
  echo "Usage: $0 [-n iter] [-s number of stream] [-p prog] [-u usr/pswd]
-h"
  echo ""
  echo "-n iter            : number of iterations, default is 100"
  echo "-s number of stream : number of streams, default is 2"
  echo "-p prog            : program to run, default is atranspl.ott"
  echo "-u usr/pswd        : user/password for database access, default is
tpcd/tpcd"
  echo "-t chkpt           : time after the start of ACID transaction to
perform the checkpoint"
  echo "                     default is 10 seconds"
  echo "-h                 : print this usage summary"
  exit 1;
}

set -- `getopt "n:p:u:s:h" "$@"` || usage

while :
do
  case "$1" in
  -s) shift; STREAM=$1;;
  -n) shift; ITER=$1;;
```


© 2004 Hewlett Packard Company. All rights reserved

```
  -p) shift; PROG=$1;;
  -u) shift; USER=$1;;
  -t) shift; CK=$1;;
  -h) usage; exit 0;;
  --) break;;
  esac
  shift
done


if [ $ITER -lt 100 ]
then
echo "Error: Must at least run 100 iterations!"
echo "Exiting..."
exit 1
fi

if [ $STREAM -lt 2 ]
then
echo "Error: Must at least run 2 streams!"
echo "Exiting..."
exit 1
fi

echo "Starting Consistency Test at `date`..."
echo ""
echo "Generate some keys first"
echo ""

i=0

while [ $i -lt $STREAM ]
do
  echo randkey $ITER 1 u$USER
  randkey $ITER 1 u$USER > ${KEY}$i
  i=`expr $i + 1`
done

echo "Check consistency before Submitting Transactions `date`"
echo "Check consistency before Submitting Transactions `date`" >>
$CON1

echo "Obtain 10 keys from the each key file to check consistency"

i=0
while [ $i -lt $STREAM ]
do
KEYS=`head -10 ${KEY}${i} | awk '{printf "%d ", $1}'`
echo "The 10 Keys for file $i are: $KEYS"
#for j in `head -10 ${KEY}${i} | awk '{printf "%d ", $1}'`
for j in $KEYS
do
   sqlplus $USER @consist $j >> $CON1
   echo "-----------------------------" >> $CON1
done
   i=`expr $i + 1`
done

echo ""
echo "Starting ACID transactions at `date`"
echo ""

i=0

while [ $i -lt $STREAM ]
do
   $PROG $i $STREAM 1 0 u${USER} i${KEY}${i}
o${OUTFILE}${i} s1 &
   i=`expr $i + 1`
done

echo "Schedule a Checkpoint"
echo "Checkpoint scheduled at $CK seconds after `date`"

(sleep $CK; $ACID_DIR/ckpt.sh) &

wait

echo ""
echo "Ending ACID transactions at `date`"
echo ""

echo "Completed $STREAM transaction streams with $ITER iterations
each"
echo ""

echo "Check consistency after Submitting Transactions `date`"
echo "Check consistency after Submitting Transactions `date`" >>
$CON2

cat ${ORACLE_HOME}/rdbms/log/alert_${ORACLE_SID}.log >>
$CHK

i=0
while [ $i -lt $STREAM ]
do
KEYS=`head -10 ${KEY}$i | awk '{printf "%d ", $1}'`
#for j in `head -10 ${KEY}$i | awk '{printf "%d ", $1}'`
echo "The keys to check for consistency after the test from file $i are:"
echo "$KEYS"
for j in $KEYS
do
   sqlplus $USER @consist $j >> $CON2
   echo "-----------------------------" >> $CON2
done
   i=`expr $i + 1`
done
```

---

consist.sql

```
Rem
Rem $Header: consist.sql 08-aug-99.16:59:17 mpoess Exp $
Rem
Rem consist.sql
Rem
Rem  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
Rem
Rem    NAME
Rem      consist.sql - <one-line expansion of the name>
Rem
Rem  DESCRIPTION
Rem       Verifies the consistency of TPC-D database using the
Rem       consistency condition.
Rem
Rem Usage:   sqlplus tpcd/tpcd @consist
Rem
Rem NOTE
Rem REQUIRES PACKAGES prvtotpt and dbmsotpt
rem
Rem    MODIFIED   (MM/DD/YY)
Rem    mpoess      08/08/99 - Creation
Rem    mpoess      08/08/99 - Created
Rem

set verify off
rem set termout on
rem set echo on
```


© 2004 Hewlett Packard Company. All rights reserved

```
REM
REM Get today's date.
REM

select
substr(TO_CHAR(sysdate,'YYYY-MM-DD HH:MI:SS'),1,20) as
CURRENT_TIME
from dual;

set serverout on;

DECLARE
        o_okey      number;
        o_tprice    number;
        l_tprice    number;
        diff                number;
BEGIN
   select o_totalprice
        into o_tprice
        from orders
        where o_orderkey = &&1;


   select sum(trunc((trunc((l_extendedprice * (1-l_discount)), 2)
      * (1+l_tax)), 2))
        into l_tprice
        from lineitem
        where l_orderkey = &&1;

   diff := l_tprice - o_tprice;

   dbms_output.put_line('O_TOTALPRICE: ' ||
TO_CHAR(trunc(o_tprice,2)));
   dbms_output.put_line('L_TOTALPRICE: ' ||
TO_CHAR(trunc(l_tprice,2)));
   dbms_output.put_line('Difference:   ' || TO_CHAR(trunc(diff,2)));

END;
.
/


spool off
exit
```

_____

dura.sh

_____

```
#!/bin/ksh
#
# $Header: dura.sh 08-aug-99.15:21:38 mpoess Exp $
#
# dura.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     dura.sh - <one-line expansion of the name>
#
#   DESCRIPTION
#     <short description of component this file declares/defines>
#
#   NOTES
#     <other useful comments, qualifications, etc.>
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/08/99 - Creation
#   mpoess      08/08/99 - Creation
#

. $KIT_DIR/env

# Create history table

# Count number of entries in the history table

SERVER="ultraperf2"

echo "-----------------------------------------------------"
echo "Capturing Process information before durability tests `date`"
rsh $SERVER -n -l spyda  ps -ef; date
echo "-----------------------------------------------------"

echo "-----------------------------------------------------"
echo "Starting the durability tests `date`"
run_acid.sh &
echo "-----------------------------------------------------"

sleep 1200

echo "-----------------------------------------------------"
echo "Collecting user information. `date`"
./cnt_user.sh pswong spyda ultraperf2 > dura/duraucnt 2>&1
echo "-----------------------------------------------------"

echo "-----------------------------------------------------"
echo "Capturing Process information while running Transactions `date`"
rsh $SERVER -n -l spyda  ps -ef; date
echo "-----------------------------------------------------"

echo "-----------------------------------------------------"
echo "Capturing disk information on Server: Ultraperf2 `date`"
rsh $SERVER -n -l spyda  vxprint -ht ; date
echo "-----------------------------------------------------"

echo "-----------------------------------------------------"
echo "Detaching mirror on data disk. `date`"
rsh $SERVER -n -l root  "vxplex -v ordr23 det ordr23-01"
echo "-----------------------------------------------------"

echo "-----------------------------------------------------"
echo "Capturing Disk information information on Server: Ultraperf2
`date`"
rsh $SERVER -n -l spyda  vxprint -ht ; date
echo "-----------------------------------------------------"

sleep 120

echo "-----------------------------------------------------"
echo "Capturing Process information after breaking data mirror. `date`"
rsh $SERVER -n -l spyda  ps -ef; date
echo "-----------------------------------------------------"


echo "-----------------------------------------------------"
echo "Detaching mirror on log2 disk. `date`"
rsh $SERVER -n -l root "vxplex -v log2 det log2-01"
echo "-----------------------------------------------------"

echo "-----------------------------------------------------"
echo "Capturing Disk information information on Server: Ultraperf2
`date`"
rsh $SERVER -n -l spyda  vxprint -ht ; date
echo "-----------------------------------------------------"

sleep 120
```


© 2004 Hewlett Packard Company. All rights reserved

```
echo "----------------------------------------------------"
echo "Capturing Process information after detaching log mirror. `date`"
rsh $SERVER -n -l spyda  ps -ef; date
echo "----------------------------------------------------"

# Power Off
```

end_acid.sh

```
#!/bin/ksh
#
# $Header: end_acid.sh 08-aug-99.17:06:20 mpoess Exp $
#
# end_acid.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     end_acid.sh - <one-line expansion of the name>
#
#  DESCRIPTION
#     end_cons.sh <pid of the durability run>
#     Options: See usage below
#
#   NOTES
#     <other useful comments, qualifications, etc.>
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/08/99 - Creation
#   mpoess      08/08/99 - Creation
#

. $KIT_DIR/env

OH=$ORACLE_HOME
# ACID_DIR=$OH/tpcd/audit set in env
OUT_DIR=$ACID_OUT/
DURA_DIR=$ACID_OUT/dura
RUN_ID_FILE=$ACID_DIR/run_id

SHELL_PID=`cat ${DURA_DIR}/shellpid`
ITER=100
STEM=${NUM_STREAMS}
let STEM="$STEM + 1" # add one for the update stream
PROG=${ACID_DIR}/atranspl.ott
IN=${ACID_DIR}/acid_in
DURA=${DURA_DIR}/drate
OUT=${DURA_DIR}/drate
DSMPL=${DURA_DIR}/durasmpl
KEY=${DURA_DIR}/key${SHELL_PID}_
USER=tpch/tpch
TRIG=1
HCNT=duracnta

# get history count

sqlplus $USER @cnt_hist > $DURA_DIR/$HCNT 2>&1

# perform the consistency

i=0
while [ $i -lt $STEM ]
do
  for j in `head -10 ${KEY}${i} | awk '{printf "%d ",$1}'`
  do
    sqlplus tpch/tpch @consist $j >> $DURA_DIR/duraconsa
  done
  i=`expr $i + 1`
done

i=0
while [ $i -lt $STEM ]
do
sample.sh $DURA${i} > ${DSMPL}${i} 2>&1
i=`expr $i + 1`
done
```

gettime.c

```
#ifdef RCSID
static char *RCSid =
  "$Header: gettime.c 15-jul-99.14:27:44 mpoess Exp $ ";
#endif /* RCSID */

/* Copyright (c) Oracle Corporation 1999. All Rights Reserved. */

/*

  NAME
    gettime.c

  DESCRIPTION
    get wall clock time.
    get cpu time.

  FUNCTIONS
    get wall clock time.
    get cpu time.

  NOTES
    Both routines return time in seconds as a double.
  MODIFIED   (MM/DD/YY)
  mpoess      07/15/99 - Creation
  mpoess      07/15/99 - Creation

*/


/*
**  Options:
**     TIME_W_TIMES:      implement gettime() with times().
**     TIME_W_GETTIME:    implement gettime() with gettimeofday().
**     CPU_W_TIMES:       implement getcpu()  with times().
**     CPU_W_GETRU:       implement getcpu()  with getrusage().
**     GETRU_STATS:       collect getrusage statistics
**     GET_P_STATS:       collect get_process_stats statistics
*/

#define SUN_OS5

#if defined(SUN_OS5)
#define TIME_W_GETTIME
#define CPU_W_TIMES
#undef GETRU_STATS
#undef CPU_W_GETRU
#endif /* SUN_OS5 */

#if defined(sequent) || defined(SEQ_PSX)
# define GET_P_STATS
#endif /* sequent */

#if defined(aix) || defined(AIXRIOS)
# define TIME_W_GETTIME
# define CPU_W_TIMES
```


© 2004 Hewlett Packard Company. All rights reserved

```
# define GETRU_STATS
#endif /* AIXRIOS */

#if defined(a_osf) || defined(A_OSF)
# define TIME_W_GETTIME
# define CPU_W_GETRU
# define GETRU_STATS
#endif /* AIXRIOS */

#if defined(HPUX) || defined(XENIX_386) || defined(SYSV_386) ||
defined(ATT_3B)
# define TIME_W_TIMES
# define CPU_W_TIMES
#endif /* HPUX || XENIX_386 || SYSV_386 */

#if !defined(TIME_W_GETTIME) && !defined(TIME_W_TIMES)
# define TIME_W_TIMES
#endif

#if !defined(CPU_W_GETRU) && !defined(CPU_W_TIMES)
# define CPU_W_TIMES
#endif

#ifdef GET_P_STATS
# ifdef GETRU_STATS
#  undef GETRU_STATS
# endif
#endif

#if defined(TIME_W_GETTIME) || defined(CPU_W_GETRU) ||
defined(GETRU_STATS)
# include <sys/time.h>
#endif /* TIME_W_GETTIME || CPU_W_GETRU || GETRU_STATS
*/

#if defined(CPU_W_GETRU) || defined(GETRU_STATS)
# include <sys/resource.h>
#endif /* CPU_W_GETRU || GETRU_STATS */

#if defined(TIME_W_TIMES) || defined (CPU_W_TIMES)
# include <sys/types.h>
# include <sys/times.h>
# include <sys/param.h>  /* most systems define HZ here */
#endif /* TIME_W_TIMES or CPU_W_TIMES */

#ifdef GET_P_STATS
# include <sys/types.h>
# include <sys/procstats.h>
#endif /* GET_P_STATS */

# include <stdio.h>

#ifdef GETRU_STATS
struct rusage selfru;
struct rusage kidsru;
#endif /* GETRU_STATS */

#ifdef GET_P_STATS
struct process_stats selfru;
struct process_stats kidsru;
#endif /* GET_P_STATS */


double gettime ()

{

#ifdef TIME_W_GETTIME
  struct timeval tv;

  (void) gettimeofday (&tv, (struct timezone *) 0);
  return ((double) tv.tv_sec + (1.0e-6 * (double) tv.tv_usec));
#endif /* TIME_W_GETTIME */

#ifdef TIME_W_TIMES
  struct tms buf;

  return ((double) times (&buf) / HZ);
#endif /* TIME_W_TIMES */

}


double getcpu ()

{

#ifdef CPU_W_TIMES
  struct tms buf;

  (void) times (&buf);
  return (((double) buf.tms_utime + (double) buf.tms_stime) / HZ);
#endif /* CPU_W_TIMES */

#ifdef CPU_W_GETRU
  struct rusage ru;
  double usecs;

  (void) getrusage (0, &ru);
  usecs = 1.0e-6 * (double) (ru.ru_utime.tv_usec + ru.ru_stime.tv_usec);
  return ((double) (ru.ru_utime.tv_sec + ru.ru_stime.tv_sec) + usecs);
#endif /* CPU_W_GETRU */

}


getru (fp, kids, config, runname, proc_no)

FILE *fp;
int kids;
char *config;
char *runname;
int proc_no;

{

#ifdef GETRU_STATS
  struct rusage ru;

  fprintf (fp, "%-10.10s %-10.10s %10d %10d ", config,runname,
proc_no, kids);
  getrusage (kids ? RUSAGE_CHILDREN : RUSAGE_SELF, &ru);
  print_ru (fp, &ru);
  fprintf (fp, "\n");
#endif /* GETRU_STATS */

#ifdef GET_P_STATS
  timeval_t tv;
  struct process_stats ru;

  fprintf (fp, "%-10.10s %-10.10s %10d %10d ", config,runname,
proc_no, kids);
  if (kids)
    get_process_stats (&tv, PS_SELF, (struct process_stats *) 0, &ru);
  else
    get_process_stats (&tv, PS_SELF, &ru, (struct process_stats *) 0);
  print_ru (fp, &ru);
```


© 2004 Hewlett Packard Company. All rights reserved

```
  fprintf (fp, "\n");
#endif /* GET_P_STATS */

}



getru1 (kids)

int kids;

{

#ifdef GETRU_STATS
  if (kids) {
    memset (&kidsru, 0, sizeof (kidsru));
    getrusage (RUSAGE_CHILDREN, &kidsru);
  }
  else {
    memset (&selfru, 0, sizeof (selfru));
    getrusage (RUSAGE_SELF, &selfru);
  }
#endif /* GETRU_STATS */

#ifdef GET_P_STATS
  timeval_t tv;

  if (kids) {
    memset (&kidsru, 0, sizeof (kidsru));
    get_process_stats (&tv, PS_SELF, (struct process_stats *) 0,
&kidsru);
  }
  else {
    memset (&selfru, 0, sizeof (selfru));
    get_process_stats (&tv, PS_SELF, &selfru, (struct process_stats *)
0);
  }
#endif /* GET_P_STATS */

}



getru2 (fp, kids, config, runname, proc_no)

FILE *fp;
int kids;
char *config;
char *runname;
int proc_no;

{

#ifdef GETRU_STATS
  struct rusage ru;

  fprintf (fp, "%-10.10s %-10.10s %10d %10d ", config, runname,
proc_no, kids);
  getrusage (kids ? RUSAGE_CHILDREN : RUSAGE_SELF, &ru);
  if (kids)
    diffru (&ru, &kidsru);
  else
    diffru (&ru, &selfru);
  print_ru (fp, &ru);
  fprintf (fp, "\n");
#endif /* GETRU_STATS */

#ifdef GET_P_STATS
  timeval_t tv;
  struct process_stats ru;

  fprintf (fp, "%-10.10s %-10.10s %10d %10d ", config, runname,
proc_no, kids);
  if (kids)
    get_process_stats (&tv, PS_SELF, (struct process_stats *) 0, &ru);
  else
    get_process_stats (&tv, PS_SELF, &ru, (struct process_stats *) 0);
  if (kids)
    diffru (&ru, &kidsru);
  else
    diffru (&ru, &selfru);
  print_ru (fp, &ru);
  fprintf (fp, "\n");
#endif /* GET_P_STATS */

}



#ifdef GETRU_STATS

print_ru (fp, ru)

FILE *fp;
struct rusage *ru;

{

  fprintf (fp, "%10ld ", ru->ru_utime.tv_sec * 1000 +
                 (ru->ru_utime.tv_usec/1000));
  fprintf (fp, "%10ld ", ru->ru_stime.tv_sec * 1000 +
                 (ru->ru_stime.tv_usec/1000));
  fprintf (fp, "%10ld ", ru->ru_maxrss);
  fprintf (fp, "%10ld ", ru->ru_majflt);
  fprintf (fp, "%10ld ", ru->ru_minflt);
  fprintf (fp, "%10ld ", 0);
  fprintf (fp, "%10ld ", 0);
  fprintf (fp, "%10ld ", 0);
  fprintf (fp, "%10ld ", ru->ru_nswap);
  fprintf (fp, "%10ld ", 0);
  fprintf (fp, "%10ld ", ru->ru_nvcsw);
  fprintf (fp, "%10ld ", ru->ru_nivcsw);
  fprintf (fp, "%10ld ", ru->ru_nsignals);
  fprintf (fp, "%10ld ", 0);
  fprintf (fp, "%10ld ", 0);
  fprintf (fp, "%10ld ", ru->ru_inblock);
  fprintf (fp, "%10ld ", ru->ru_oublock);
  fprintf (fp, "%10ld ", 0);
  fprintf (fp, "%10ld",  0);

}



diffru (ru2, ru)

struct rusage *ru2;
struct rusage *ru;

{

  ru2->ru_utime.tv_sec -= ru->ru_utime.tv_sec;
  ru2->ru_utime.tv_usec -= ru->ru_utime.tv_usec;
  ru2->ru_stime.tv_sec -= ru->ru_stime.tv_sec;
  ru2->ru_stime.tv_usec -= ru->ru_stime.tv_usec;
  ru2->ru_maxrss -= ru->ru_maxrss;
  ru2->ru_ixrss -= ru->ru_ixrss;
  ru2->ru_idrss -= ru->ru_idrss;
  ru2->ru_minflt -= ru->ru_minflt;
  ru2->ru_majflt -= ru->ru_majflt;
```


© 2004 Hewlett Packard Company. All rights reserved

```
  ru2->ru_nswap -= ru->ru_nswap;
  ru2->ru_inblock -= ru->ru_inblock;
  ru2->ru_oublock -= ru->ru_oublock;
  ru2->ru_msgsnd -= ru->ru_msgsnd;
  ru2->ru_msgrcv -= ru->ru_msgrcv;
  ru2->ru_nsignals -= ru->ru_nsignals;
  ru2->ru_nvcsw -= ru->ru_nvcsw;
  ru2->ru_nivcsw -= ru->ru_nivcsw;

}

#endif /* GETRU_STATS */



#ifdef GET_P_STATS

print_ru (fp, ps)

FILE *fp;
struct process_stats *ps;

{

  fprintf (fp, "%lu ", ps->ps_utime.tv_sec * 1000 +
                 (ps->ps_utime.tv_usec/1000));
  fprintf (fp, "%lu ", ps->ps_stime.tv_sec * 1000 +
                 (ps->ps_stime.tv_usec/1000));
  fprintf (fp, "%lu ", ps->ps_maxrss);
  fprintf (fp, "%lu ", ps->ps_pagein);
  fprintf (fp, "%lu ", ps->ps_reclaim);
  fprintf (fp, "%lu ", ps->ps_zerofill);
  fprintf (fp, "%lu ", ps->ps_pffincr);
  fprintf (fp, "%lu ", ps->ps_pffdecr);
  fprintf (fp, "%lu ", ps->ps_swap);
  fprintf (fp, "%lu ", ps->ps_syscall);
  fprintf (fp, "%lu ", ps->ps_volcsw);
  fprintf (fp, "%lu ", ps->ps_involcsw);
  fprintf (fp, "%lu ", ps->ps_signal);
  fprintf (fp, "%lu ", ps->ps_lread);
  fprintf (fp, "%lu ", ps->ps_lwrite);
  fprintf (fp, "%lu ", ps->ps_bread);
  fprintf (fp, "%lu ", ps->ps_bwrite);
  fprintf (fp, "%lu ", ps->ps_phread);
  fprintf (fp, "%lu",  ps->ps_phwrite);

}



diffru (ru2, ru)

struct process_stats *ru2;
struct process_stats *ru;

{

  ru2->ps_utime.tv_sec -= ru->ps_utime.tv_sec;
  ru2->ps_utime.tv_usec -= ru->ps_utime.tv_usec;
  ru2->ps_stime.tv_sec -= ru->ps_stime.tv_sec;
  ru2->ps_stime.tv_usec -= ru->ps_stime.tv_usec;
  ru2->ps_maxrss -= ru->ps_maxrss;
  ru2->ps_pagein -= ru->ps_pagein;
  ru2->ps_reclaim -= ru->ps_reclaim;
  ru2->ps_zerofill -= ru->ps_zerofill;
  ru2->ps_pffincr -= ru->ps_pffincr;
  ru2->ps_pffdecr -= ru->ps_pffdecr;
  ru2->ps_swap -= ru->ps_swap;
  ru2->ps_syscall -= ru->ps_syscall;
  ru2->ps_volcsw -= ru->ps_volcsw;
  ru2->ps_involcsw -= ru->ps_involcsw;
  ru2->ps_signal -= ru->ps_signal;
  ru2->ps_lread -= ru->ps_lread;
  ru2->ps_lwrite -= ru->ps_lwrite;
  ru2->ps_bread -= ru->ps_bread;
  ru2->ps_bwrite -= ru->ps_bwrite;
  ru2->ps_phread -= ru->ps_phread;
  ru2->ps_phwrite -= ru->ps_phwrite;

}

#endif /* GET_P_STATS */
```

---

gtime.c

```
/* Copyright (c) 2001, 2002, Oracle Corporation.  All rights reserved.  */

/*

  NAME
    gtime.c - <one-line expansion of the name>

  DESCRIPTION
    <short description of facility this file declares/defines>

  EXPORT FUNCTION(S)
    <external functions defined for use outside package - one-line
descriptions>

  INTERNAL FUNCTION(S)
    <other external functions defined - one-line descriptions>

  STATIC FUNCTION(S)
    <static functions defined - one-line descriptions>

  NOTES
    <other useful comments, qualifications, etc.>

  MODIFIED   (MM/DD/YY)
  mpoess      10/23/02 - mpoess_update_from_visa
  mpoess      08/29/01 - Creation

*/
#include<stdio.h>
#include<stdlib.h>

# include <sys/time.h>

main ()
{

  struct timeval tv;

       (void) gettimeofday (&tv, (struct timezone *) 0);

  printf ("%.2f\n", ((double) tv.tv_sec + (1.0e-6 * (double) tv.tv_usec)) )
;

}

/* end of file gtime.c */
```

---

iso1.sh

```
#!/bin/ksh
```


© 2004 Hewlett Packard Company. All rights reserved

```
#
# $Header: iso1.sh 29-jul-98.17:00:11 akarasik Exp $
#
# iso1.sh
#
#  Copyright (c) Oracle Corporation 1998. All Rights Reserved.
#
#   NAME
#     iso1.sh
#
#  DESCRIPTION
#      Usage: iso1.sh [-u user/password] [-n remote_node] -h
#      Options: See usage below
#  NOTES
#             For a cross node isolation test, assume the local node is
#             one of the participating nodes.  The other node can be
#             specified by the -n option.
#      You need to set the environment variable TPCD_KIT_DIR
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      12/16/98 - update to version 8.1.6
#   mpoess      09/25/98 - update audit
#   akarasik    07/29/98 -
#   akarasik    07/29/98 - Creation
#

. $KIT_DIR/env

# May need to change the following:
RSH=rsh

OH=$ORACLE_HOME
#ACID_DIR=$KIT_DIR/acid is set in env
OUT_DIR=$ACID_OUT

TXN1FILE=$OUT_DIR/txn1$$.out
TXN2FILE=$OUT_DIR/txn2$$.out
KEYFILE=$OUT_DIR/key$$.out
ISOFILE=$OUT_DIR/iso1

USER=$DATABASE_USER
PROG=atranspl


/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE

trap "/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE; exit 1" 1 2 3 15

usage() {

  echo ""
  echo "Usage: $0 [-u user/passwd] [-n remote_node] -h"
  echo ""
  exit 1;
}

set -- `getopt "u:n:h" "$@"` || usage

while :
do
  case "$1" in
  -u) shift; USER=$1;;
  -n) shift; HOST="$1";;
  -h) usage; exit 0;;
  --) break;;
  esac
  shift;
done
de=`direxists.sh $ACID_OUT c` # I am not using $de afterward, but I
want to avoid the output of direxists

# generate key files

randkey 1 0.1 u"$USER" > $KEYFILE

OKEY=`cat $KEYFILE | awk '{print $1}'`
echo "o_key is "$OKEY

# before the ACID transaction, let's run a ACID query to record the
# initial state of lineitem

echo "Running ACID query BEFORE the start of Isolation Test 1" >>
$TXN2FILE
echo "`date`" >> $TXN2FILE
echo "" >> $TXN2FILE
sqlplus $USER @$ACID_DIR/isolation/a_query $OKEY >>
$TXN2FILE
echo "" >> $TXN2FILE
echo "-----------------------------------------------" >> $TXN2FILE

sleep 1

# start ACID transaction, Sleep for 60 second before COMMIT

$PROG 1 1 1 0 i$KEYFILE u$USER s60 b0 >> $TXN1FILE &

# let's sleep 10 seconds before starting ACID query

sleep 15

# start ACID query with the same OKEY

echo "Running ACID query 15 seconds AFTER the start of ACID
Transaction" \
>> $TXN2FILE
echo ${RSH} -n ${HOST}  sqlplus $USER
@$ACID_DIR/isolation/a_query $OKEY
echo "`date`" >> $TXN2FILE
if [ "$HOST" != "" ]
then
echo "Starting ACID query on node $HOST" >> $TXN2FILE
echo ${RSH} -n ${HOST}  sqlplus $USER
@$ACID_DIR/isolation/a_query $OKEY
${RSH} -n ${HOST}  sqlplus $USER @$ACID_DIR/isolation/a_query
$OKEY >> $TXN2FILE
else
sqlplus $USER @$ACID_DIR/isolation/a_query $OKEY >>
$TXN2FILE
fi

echo "-----------------------------------------------" >> $TXN2FILE
wait
echo "-----------------------------------------------" >> $TXN1FILE

cat $TXN1FILE $TXN2FILE >> $ISOFILE

/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE
```

iso2.sh

```
#!/bin/ksh
#
# $Header: iso2.sh 04-aug-99.09:19:54 mpoess Exp $
#
# iso2.sh
```


© 2004 Hewlett Packard Company. All rights reserved

```
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     iso2.sh - <one-line expansion of the name>
#
#   DESCRIPTION
#     Usage: iso2.sh [-u user/password] [-n remote_node] -h
#     Options: See usage below
#   NOTES
#      For a cross node isolation test, assume the local node is
#      one of the participating nodes.  The other node can be
#      specified by the -n option.
#      You need to set the environment variable TPCD_KIT_DIR
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/04/99 - Creation
#   mpoess      08/04/99 - Creation
#
#
#
=================================================
===============+
# May need to change the following:

. $KIT_DIR/env

RSH=rsh

OH=$ORACLE_HOME
# ACID_DIR=$TPCD_KIT_DIR/audit is set in env
OUT_DIR=$ACID_OUT

DURA_DIR=$ACID_DIR/dura

TXN1FILE=$OUT_DIR/txn1$$.out
TXN2FILE=$OUT_DIR/txn2$$.out
KEYFILE=$OUT_DIR/key$$.out
ISOFILE=$OUT_DIR/iso2

USER=$DATABASE_USER
PROG=atranspl

/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE

trap "/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE; exit 1" 1 2 3 15

usage() {

  echo ""
  echo "Usage: $0 [-u user/passwd] [-n remote_node] -h"
  echo ""
  exit 1;
}

set -- `getopt "u:n:h" "$@"` || usage

while :
do
  case "$1" in
  -u) shift; USER=$1;;
  -n) shift; HOST="$1";;
  -h) usage; exit 0;;
  --) break;;
  esac
  shift;
done


# generate key files

randkey 1 0.1 u"$USER" > $KEYFILE

OKEY=`cat $KEYFILE | awk '{print $1}'`
echo "o_key is "$OKEY

# before the ACID transaction, let's run a ACID query to record the
# initial state of lineitem

echo "Running ACID query BEFORE the start of Isolation Test 1" >>
$TXN2FILE
echo "`date`" >> $TXN2FILE
echo "" >> $TXN2FILE
sqlplus "$USER" @$ACID_DIR/isolation/a_query $OKEY >>
$TXN2FILE
echo "" >> $TXN2FILE
echo "-----------------------------------------------" >> $TXN2FILE

sleep 1

# start ACID transaction, Sleep for 30 second before ROLLBACK

$PROG 1 1 0 0 i$KEYFILE u$USER s30 >> $TXN1FILE &

# let's sleep 15 seconds before starting ACID query

sleep 15

# start ACID query with the same OKEY

echo "Running ACID query 15 seconds AFTER the start of ACID
transaction" \
>> $TXN2FILE
echo "`date`" >> $TXN2FILE
if [ "$HOST" != "" ]
then
echo "Starting ACID query on node $HOST" >> $TXN2FILE
${RSH} -n ${HOST}  sqlplus "$USER"
@$ACID_DIR/isolation/a_query $OKEY >> $TXN2FILE
else
sqlplus $USER @$ACID_DIR/isolation/a_query $OKEY >>
$TXN2FILE
fi

echo "-----------------------------------------------" >> $TXN2FILE
wait
echo "-----------------------------------------------" >> $TXN1FILE

cat $TXN1FILE $TXN2FILE >> $ISOFILE

/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE
```

_____

iso3.sh

_____

```
#!/bin/ksh
#
# $Header: iso3.sh 04-aug-99.09:20:35 mpoess Exp $
#
# iso3.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     iso3.sh - <one-line expansion of the name>
#
#  DESCRIPTION
#     Usage: iso3.sh [-u user/password] [-n remote_node] -h
#     Options: See usage below
```


© 2004 Hewlett Packard Company. All rights reserved

```
#  NOTES
#      For a cross node isolation test, assume the local node is
#      one of the participating nodes.  The other node can be
#      specified by the -n option.
#              We need to make sure the remote node has access to the
#              file system on the local node.  Otherwise, we need to rcp
#              the keyfile to the remote system.
#      You need to set the environment variable TPCD_KIT_DIR
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/04/99 - Creation
#   mpoess      08/04/99 - Creation
#

. $KIT_DIR/env

# May need to change the following:
RSH=rsh

OH=$ORACLE_HOME
#ACID_DIR=$TPCD_KIT_DIR/audit is set in env
OUT_DIR=$ACID_OUT

DURA_DIR=$ACID_DIR/dura

TXN1FILE=$OUT_DIR/txn1$$.out
TXN2FILE=$OUT_DIR/txn2$$.out
KEYFILE=$OUT_DIR/key$$.out
ISOFILE=$OUT_DIR/iso3

USER=$DATABASE_USER
PROG=$KIT_DIR/utils/atranspl

/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE

trap "/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE; exit 1" 1 2 3 15

usage() {

  echo ""
  echo "Usage: $0 [-u user/passwd] [-n remote_node] -h"
  echo ""
  exit 1;
}

set -- `getopt "u:n:h" "$@"` || usage

while :
do
  case "$1" in
  -u) shift; USER=$1;;
  -n) shift; HOST="$1";;
  -h) usage; exit 0;;
  --) break;;
  esac
  shift
done


# generate key files

randkey 1 0.1 u"$USER" > $KEYFILE
echo Host $HOST
rcp $KEYFILE ${HOST}:$KEYFILE

sleep 1

# start ACID transaction, Sleep for 30 second before COMMIT

$PROG 1 2 1 0 i$KEYFILE u$USER s60 b0 >> $TXN1FILE &

# let's sleep 15 seconds before starting second ACID transaction

sleep 15

# start another ACID transaction with the same LKEY and OKEY
# but different DELTA

# Do not sleep before COMMIT so that we can see TXN2 has waited.

if [ "$HOST" != "" ]
then
echo "Starting TXN2 on node $HOST" >> $TXN2FILE
${RSH} -n ${HOST} $PROG 2 2 1 1 i$KEYFILE u$USER s1 b1 >>
$TXN2FILE &
else
$PROG 2 2 1 1 i$KEYFILE u$USER s1 b1 >> $TXN2FILE &
fi

wait
echo "-----------------------------------------------" >> $TXN2FILE
echo "-----------------------------------------------" >> $TXN1FILE

cat $TXN1FILE $TXN2FILE >> $ISOFILE

/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE
```

iso4.sh

```
#!/bin/ksh
#
# $Header: iso4.sh 04-aug-99.09:21:12 mpoess Exp $
#
# iso4.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     iso4.sh - <one-line expansion of the name>
#
#  DESCRIPTION
#     Usage: iso4.sh [-u user/password] [-n remote_node] -h
#     Options: See usage below
#  NOTES
#      For a cross node isolation test, assume the local node is
#      one of the participating nodes.  The other node can be
#      specified by the -n option.
#      We need to make sure the remote node has access to the
#      file system on the local node.  Otherwise, we need to rcp
#      the keyfile to the remote system.
#      You need to set the environment variable TPCD_KIT_DIR
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/04/99 - Creation
#   mpoess      08/04/99 - Creation
#

. $KIT_DIR/env

# May need to change the following:
RSH=rsh

OH=$ORACLE_HOME
# ACID_DIR=$TPCD_KIT_DIR/audit is set in env
OUT_DIR=$ACID_OUT

DURA_DIR=$ACID_DIR/dura
```


© 2004 Hewlett Packard Company. All rights reserved

```
TXN1FILE=$OUT_DIR/txn1$$.out
TXN2FILE=$OUT_DIR/txn2$$.out
KEYFILE=$OUT_DIR/key$$.out
ISOFILE=$OUT_DIR/iso4

USER=$DATABASE_USER
PROG=$KIT_DIR/utils/atranspl

/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE

trap "/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE; exit 1" 1 2 3 15

usage() {

  echo ""
  echo "Usage: $0 [-u user/passwd] [-n remote_node] -h"
  echo ""
  exit 1;
}

set -- `getopt "u:n:h" "$@"` || usage

while :
do
  case "$1" in
  -u) shift; USER=$1;;
  -n) shift; HOST="$1";;
  -h) usage; exit 0;;
  --) break;;
  esac
  shift
done


# generate key files

randkey 1 0.1 u"$USER" > $KEYFILE
rcp $KEYFILE ${HOST}:$KEYFILE

sleep 1

# start ACID transaction, Sleep for 30 second before ROLLBACK

$PROG 1 2 0 0 i$KEYFILE u$USER s30 b0 >> $TXN1FILE &

# let's sleep 15 seconds before starting second ACID transaction

sleep 15

# start another ACID transaction with the same LKEY and OKEY
# but different DELTA

# Do not sleep before COMMIT so that we can see TXN2 has waited.

if [ "$HOST" != "" ]
then
echo "Starting TXN2 on node $HOST" >> $TXN2FILE
${RSH} -n ${HOST} $PROG 2 2 1 1 i$KEYFILE u$USER s1 b1 >>
$TXN2FILE &
else
$PROG 2 2 1 1 i$KEYFILE u$USER s1 b1 >> $TXN2FILE &
fi

wait
echo "-----------------------------------------------" >> $TXN2FILE
echo "-----------------------------------------------" >> $TXN1FILE

cat $TXN1FILE $TXN2FILE >> $ISOFILE

/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE
```

iso5.sh

```
#!/bin/ksh
#
# $Header: iso5.sh 04-aug-99.09:21:45 mpoess Exp $
#
# iso5.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     iso5.sh - <one-line expansion of the name>
#
#  DESCRIPTION
#     Usage: iso5.sh [-u user/password] [-n remote_node] -h
#     Options: See usage below
#  NOTES
#      For a cross node isolation test, assume the local node is
#      one of the participating nodes.  The other node can be
#      specified by the -n option.
#      You need to set the environment variable TPCD_KIT_DIR
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/04/99 - Creation
#   mpoess      08/04/99 - Creation
#

. $KIT_DIR/env

# May need to change the following:
RSH=rsh

OH=$ORACLE_HOME
# ACID_DIR=$TPCD_KIT_DIR/audit is set in env
OUT_DIR=$ACID_OUT
DURA_DIR=$ACID_DIR/dura

TXN1FILE=$OUT_DIR/txn1$$.out
TXN2FILE=$OUT_DIR/txn2$$.out
KEYFILE=$OUT_DIR/key$$.out
ISOFILE=$OUT_DIR/iso5

USER=$DATABASE_USER
PROG=atranspl

/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE

trap "/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE; exit 1" 1 2 3 15

usage() {

  echo ""
  echo "Usage: $0 [-u user/passwd] [-n remote_node] -h"
  echo ""
  exit 1;
}

set -- `getopt "u:n:h" "$@"` || usage

while :
do
  case "$1" in
  -u) shift; USER=$1;;
  -n) shift; HOST="$1";;
  -h) usage; exit 0;;
  --) break;;
  esac
```


© 2004 Hewlett Packard Company. All rights reserved

```
  shift;
done


# generate key files

randkey 1 0.1 u"$USER" > $KEYFILE
rcp $KEYFILE ${HOST}:$KEYFILE

OKEY=`cat $KEYFILE | awk '{print $1}'`
echo "o_key is "$OKEY

# before the ACID transaction, let's run a ACID query to record the
# initial state of lineitem

echo "Running ACID query BEFORE the start of Isolation Test 5" >>
$TXN1FILE
echo "`date`" >> $TXN1FILE
echo "" >> $TXN1FILE
sqlplus $USER @$ACID_DIR/isolation/a_query $OKEY >>
$TXN1FILE
echo "" >> $TXN1FILE
echo "-----------------------------------------------" >> $TXN1FILE

sleep 1

# start ACID transaction, Sleep for 60 second before COMMIT

$PROG 1 1 1 0 i$KEYFILE u$USER s60 >> $TXN1FILE &

# let's sleep 5 seconds before starting PARTSUPP query

sleep 5

# First generate PS_PARTKEY and PS_SUPPKEY

PSKEY=`randpsup 1`

echo "Running PARTSUPP query 5 seconds AFTER the start of ACID
Transaction" \
>> $TXN2FILE
echo "`date`" >> $TXN2FILE
echo "PS_PARTKEY and PS_SUPPKEY are: $PSKEY" >>
$TXN2FILE

if [ "$HOST" != "" ]
then
echo "Starting PARTSUPP query on node $HOST" >> $TXN2FILE
${RSH} -n ${HOST} sqlplus $USER
@$ACID_DIR/isolation/a_query2 ${PSKEY} >> $TXN2FILE &
else
sqlplus $USER @$ACID_DIR/isolation/a_query2 ${PSKEY} >>
$TXN2FILE &
fi

wait

echo "-----------------------------------------------" >> $TXN2FILE
echo "-----------------------------------------------" >> $TXN1FILE

cat $TXN1FILE $TXN2FILE >> $ISOFILE

/bin/rm -rf $TXN1FILE $TXN2FILE $KEYFILE
```

_____

iso6.sh

_____

```
#!/bin/ksh
#
# $Header: iso6.sh 04-aug-99.09:22:12 mpoess Exp $
#
# iso6.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     iso6.sh - <one-line expansion of the name>
#
#  DESCRIPTION
#     Usage: iso6.sh [-u user/password] [-n remote_node] -h
#     Options: See usage below
#  NOTES
#      For a cross node isolation test, assume the local node is
#      one of the participating nodes.  The other node can be
#      specified by the -n option.
#      We need to make sure the remote node has access to the
#      file system on the local node.  Otherwise, we need to rcp
#      the keyfile to the remote system.
#      You need to set the environment variable TPCD_KIT_DIR
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/04/99 - Creation
#   mpoess      08/04/99 - Creation
#

. $KIT_DIR/env

# May need to change the following:
RSH=rsh

OH=/private/tpcd
# ACID_DIR=$TPCD_KIT_DIR/audit is set in env
OUT_DIR=$ACID_OUT

DURA_DIR=$ACID_DIR/dura

TXN1FILE=$OUT_DIR/txn1$$.out
TXN2FILE=$OUT_DIR/txn2$$.out
TXN3FILE=$OUT_DIR/txn3$$.out
KEYFILE=$OUT_DIR/key$$.out
ISOFILE=$OUT_DIR/iso6

USER=$DATABASE_USER
PROG=$KIT_DIR/utils/atranspl

/bin/rm -rf $TXN1FILE $TXN2FILE $TXN3FILE $KEYFILE

trap "/bin/rm -rf $TXN1FILE $TXN2FILE $TXN3FILE $KEYFILE;
exit 1" 1 2 3 15

usage() {

  echo ""
  echo "Usage: $0 [-u user/passwd] [-n remote_node] -h"
  echo ""
  exit 1;
}

set -- `getopt "u:n:h" "$@"` || usage

while :
do
  case "$1" in
  -u) shift; USER=$1;;
  -n) shift; HOST="$1";;
  -h) usage; exit 0;;
  --) break;;
  esac
  shift;
```


© 2004 Hewlett Packard Company. All rights reserved

```
done


# generate key files

randkey 1 0.1 u"$USER" > $KEYFILE
rcp $KEYFILE ${HOST}:$KEYFILE

OKEY=`cat $KEYFILE | awk '{print $1}'`
echo "o_key is "$OKEY

# before the any transaction, let's run a ACID query to record the
# initial state of lineitem

echo "Running ACID query BEFORE the start of Isolation Test 6" >>
$TXN2FILE
echo "`date`" >> $TXN2FILE
echo "" >> $TXN2FILE
sqlplus $USER @$ACID_DIR/isolation/a_query $OKEY >>
$TXN2FILE
echo "" >> $TXN2FILE
echo "-----------------------------------------------" >> $TXN2FILE

sleep 1

# start Query 1, use 0 as the delta

echo "Running Query 17b at `date`" >> $TXN1FILE
sqlplus $USER @q1 >> $TXN1FILE &

# sleep 2 seconds before starting ACID transaction

sleep 2

# start ACID transaction, COMMIT after one second

echo "Starting AICD transaction at `date`" >> $TXN2FILE

if [ "$HOST" != "" ]
then
echo "Starting ACID transaction on node $HOST" >> $TXN2FILE
${RSH} -n ${HOST} $PROG 1 1 1 0 i$KEYFILE u$USER s1 >>
$TXN2FILE &
else
$PROG 1 1 1 0 i$KEYFILE u$USER s1 >> $TXN2FILE &
fi

# start Query 17

sleep 2

echo "Running 2nd Query 17b at `date`" >> $TXN3FILE
sqlplus $USER @q1 >> $TXN3FILE &
# wait for everyone to finish

wait

echo "-----------------------------------------------" >> $TXN3FILE
echo "-----------------------------------------------" >> $TXN2FILE
echo "-----------------------------------------------" >> $TXN1FILE

cat $TXN1FILE $TXN2FILE $TXN3FILE >> $ISOFILE

/bin/rm -rf $TXN1FILE $TXN2FILE $TXN3FILE $KEYFILE
```

randkey.c

```
/* Copyright (c) 2001, 2002, Oracle Corporation.  All rights reserved.  */

/*

  NAME
    randkey.c - <one-line expansion of the name>

  DESCRIPTION
    Generate random keys for ACID transactions:
    O_ORDERKEY unique random (1..SF*150000*4) and only
    first 8 keys out of every 32 are populated.
      and
    L_ORDERKEY based on Clause 3.1.6.2
    DELTA random (1..100)
*/

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "atranspl.h"

#define ORDERCNT 150000.0

/* MK_SPARSE adopted from dss.h */

#define  MK_SPARSE(key, seq) \
        (((((key>>3)<<2)|(seq & 0x0003))<<3)|(key & 0x0007))

void sql_error();
void usage();
void ACIDinit();
long atol();
void srand48();
long lrand48();

/* Not really used here, but retained it for future purposes. */

typedef struct aciddef {
  long okey;
  long lkey;
  int delta;
} adef;

long l_key = 0;
long o_key = 0;
char lname[UNAME_LEN];
char *passwd;

/* OCI handles */

OCIEnv *tpcenv;
OCIServer *tpcsrv;
OCIError *errhp;
OCISvcCtx *tpcsvc;
OCISession *tpcusr;
OCIStmt *curi;

OCIBind *l_key_bp;
OCIBind *o_key_bp;

sword status = OCI_SUCCESS;  /* OCI return value */

char sqlstmt[1024];


void ACIDexit() {
  OCILogoff(tpcsvc,errhp);
  OCIhfree(tpcenv,OCI_HTYPE_STMT);
  OCIhfree(tpcsvc,OCI_HTYPE_SVCCTX);
  OCIhfree(tpcsrv,OCI_HTYPE_SERVER);
  OCIhfree(tpcusr,OCI_HTYPE_SESSION);
```


© 2004 Hewlett Packard Company. All rights reserved

```
}


/* type: 0 if environment handle is passed, 1 if error handle is passwd */

void sql_error(errhp,status,type)
    OCIError *errhp;
    sword status;
    sword type;
{
 char msg[2048];
 sb4 errcode;
 ub4 msglen;
 int i,j;

 switch(status) {
 case OCI_SUCCESS_WITH_INFO:
   fprintf(stderr, "Error: Statement returned with info.\n");
   if (type)
     (void) OCIErrorGet(errhp,1,NULL,(sb4 *) &errcode,(text *)msg,
                    2048,OCI_HTYPE_ERROR);
   else
     (void) OCIErrorGet(errhp,1,NULL,(sb4 *) &errcode,(text *)msg,
                    2048,OCI_HTYPE_ENV);
   fprintf(stderr,"%s\n",msg);
   break;
 case OCI_ERROR:
   fprintf(stderr, "Error: OCI call error.\n");
   if (type)
     (void) OCIErrorGet(errhp,1,NULL,(sb4 *) &errcode,(text *)msg,
                    2048,OCI_HTYPE_ERROR);
   else
     (void) OCIErrorGet(errhp,1,NULL,(sb4 *) &errcode,(text *)msg,
                    2048,OCI_HTYPE_ENV);
   fprintf(stderr,"%s\n",msg);
   break;
 case OCI_INVALID_HANDLE:
   fprintf(stderr, "Error: Invalid Handle.\n");
   if (type)
     (void) OCIErrorGet(errhp,1,NULL,(sb4 *) &errcode,(text *)msg,
                    2048,OCI_HTYPE_ERROR);
   else
     (void) OCIErrorGet(errhp,1,NULL,(sb4 *) &errcode,(text *)msg,
                    2048,OCI_HTYPE_ENV);
   fprintf(stderr,"%s\n",msg);
   break;
 }
 /* Rollback just in case */

 (void) OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);

 fprintf(stderr, "Exiting Oracle...\n");
 fflush(stderr);

 ACIDexit();

 exit(1);
}


main(argc, argv)
    int argc;
    char **argv;
{

 long count;
 long i;
 double sf;        /* need to accomodate sf 0.1 */
 double random;
 double ordcnt;
 adef *res;

 if ((argc < 3) || (argc > 4)) {
   usage();
   exit(-1);
 }

 strcpy((char *) lname, "tpcd/tpcd");

 count = atol(argv[1]);
 sf = atof(argv[2]);

 argc -= 2;
 argv += 2;

 while (--argc) {
   ++argv;
   switch(argv[0][0]) {
   case 'u':
     strncpy((char *) lname, ++(argv[0]), UNAME_LEN);
     if (strchr((char *) lname, '/') == NULL) {
       usage();
       exit(-1);
     }
     break;
   default:
     fprintf(stderr, "Unknown argument %s\n", argv[0]);
     usage();
     break;
   }
 }

 ACIDinit();

 /* initialize array for random numbers */

 res = (adef *) malloc(count*sizeof(adef));
 ordcnt = (double) ORDERCNT * (double) sf;

 for (i=0; i<count; i++) {

   /* The algorithm:                                    */
   /* Assumes drand's output is 'unique', first get a number within */
   /* the range of [0..sf*ORDERCNT) and then maps the different     */
   /* ranges to generate the real output.                           */

   random = floor(drand48() * (double) ordcnt) + 1;
   res[i].okey = o_key = (long) MK_SPARSE((long) random, 0);
   res[i].delta = (long) floor(drand48() * 100) + 1;

   /* Obtain l_key from l_key query */

   OCIsexec(tpcsvc,curi,errhp,1);

   /* l_key is the highest l_linenumber available.  We need to pick */
   /* at random a number between 1..l_key.                          */

   res[i].lkey = (lrand48() % l_key) + 1;

   printf("%ld %ld %d\n", res[i].okey, res[i].lkey, res[i].delta);
 }

 ACIDexit();
 free(res);

}


void usage() {
```


© 2004 Hewlett Packard Company. All rights reserved

```
  fprintf(stderr, "Usage: randkey <number of random keys to generate>
<SF> u<user/password>\n");
  fprintf(stderr, "\n");
}

void ACIDinit()
{

  /* run random seed */

  srand48(getpid());

  /* Connect to ORACLE.  Program will call sql_error()
     if an error occurs in connecting to the default database. */

  (void) OCIInitialize(OCI_DEFAULT,(dvoid *)0,0,0,0);
  if((status=OCIEnvInit((OCIEnv **)&tpcenv,OCI_DEFAULT,0,(dvoid
**)0)) !=
     OCI_SUCCESS)
    sql_error(tpcenv, status, 0);

  OCIhalloc(tpcenv,&errhp,OCI_HTYPE_ERROR);
  OCIhalloc(tpcenv,&curi,OCI_HTYPE_STMT);
  OCIhalloc(tpcenv,&tpcsvc,OCI_HTYPE_SVCCTX);
  OCIhalloc(tpcenv,&tpcsrv,OCI_HTYPE_SERVER);
  OCIhalloc(tpcenv,&tpcusr,OCI_HTYPE_SESSION);

  /* get username and password */

  passwd = strchr(lname, '/');
  *passwd = '\0';
  passwd++;

  if ((status=OCIServerAttach(tpcsrv,errhp,(text
*)0,0,OCI_DEFAULT))!=OCI_SUCCESS)
    sql_error(errhp,status,1);


OCIaset(tpcsvc,OCI_HTYPE_SVCCTX,tpcsrv,0,OCI_ATTR_SERVER
,errhp);

OCIaset(tpcusr,OCI_HTYPE_SESSION,lname,strlen(lname),OCI_ATT
R_USERNAME,
         errhp);

OCIaset(tpcusr,OCI_HTYPE_SESSION,passwd,strlen(passwd),OCI_A
TTR_PASSWORD,
         errhp);

  if ((status = OCISessionBegin(tpcsvc, errhp, tpcusr,
OCI_CRED_RDBMS,
                        OCI_DEFAULT)) != OCI_SUCCESS)
    sql_error(errhp,status,1);


OCIaset(tpcsvc,OCI_HTYPE_SVCCTX,tpcusr,0,OCI_ATTR_SESSIO
N,errhp);

  /* Open and Parse cursor for query to choose determine l_key. */
  /* Binds l_key to :l_key.                                     */

  sprintf((char *) sqlstmt,SQLTXT1);
  OCIStmtPrepare(curi,errhp,(text *)sqlstmt,strlen((char *)sqlstmt),
              OCI_NTV_SYNTAX,OCI_DEFAULT);


OCIbbname(curi,l_key_bp,errhp,":l_key",ADR(l_key),SIZ(l_key),SQL
T_INT);

OCIbbname(curi,o_key_bp,errhp,":o_key",ADR(o_key),SIZ(o_key),SQ
LT_INT);
}
```

randpsup.c

```
/* Copyright (c) 2001, 2002, Oracle Corporation.  All rights reserved.  */

/*

   NAME
     randpsup.c - <one-line expansion of the name>

   DESCRIPTION
     Generate random keys for ACID PARTSUPP transactions:
     (Clause 4.2.3)
     PS_PARTKEY random within [SF*200000]
       and
     PS_SUPPKEY = (PS_PARTKEY + (i * ((S/4) +
(int)(PS_PARTKEY - 1)
                /S))) % S + 1
     where i random within [0..3] and S = SF * 10000

MODIFIED
   mpoess      10/23/02 - mpoess_update_from_visa
   mpoess      01/04/01 - Creation

*/

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#define PS_PER_SF 200000.0
#define S_PER_SF 10000.0
#define SUPP_PER_PART 4

/* borrowed from build.c in the dbgen distribution */

#define PART_SUPP_BRIDGE(tgt, p, s) \
    { \
    long tot_scnt = (long) (S_PER_SF * sf); \
    tgt = (p + s *  (tot_scnt / SUPP_PER_PART +  \
       (long) ((p - 1) / tot_scnt))) % tot_scnt + 1; \
    }

void usage();
double atof();
void srand48();
long lrand48();


main(argc, argv)
     int argc;
     char **argv;
{

  double sf = 0.1;          /* scale factor      */
  long supp;               /* the i-th supplier */
  long pkey;               /* partkey           */
  long maxpkey;              /* highest partkey   */
  long ps_skey;             /* ps_suppkey        */

  if (argc < 2) {
    usage();
    exit(-1);
  }
```


© 2004 Hewlett Packard Company. All rights reserved

```
  /* seed the random number generator */

  srand48(getpid());

  sf = atof(argv[1]);
  maxpkey = (long) (sf * PS_PER_SF);
  supp = lrand48() % 4;
  pkey = lrand48() % maxpkey + 1;

  PART_SUPP_BRIDGE(ps_skey, pkey, supp);

  fprintf(stdout, "%ld %ld", pkey, ps_skey);

  exit(0);
}


void usage()
{
  fprintf(stderr, "Usage: randpsup <SF>\n\n");
}
```

run_acid.sh

```
#!/bin/ksh
#
# $Header: run_acid.sh 08-aug-99.15:30:10 mpoess Exp $
#
# run_acid.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     run_acid.sh - <one-line expansion of the name>
#
#  DESCRIPTION
#     Usage: run_acid.sh [-n iter] [-s stream] [-p prog] [-i infile]
#                        [-o outfile] [-d durafile] [-u usr/pswd]
#                        [-t trigger] [-f scale factor] -h
#
#     Options: See usage below
#
#   MODIFIED   (MM/DD/YY)
#   mpoess     08/08/99 - Creation
#   mpoess     08/08/99 - Creation
#

. $KIT_DIR/env

OH=$ORACLE_HOME
ACID_DIR=$ACID_DIR
OUT_DIR=$ACID_OUT

usage() {

  echo ""
  echo "Usage: $0 [-n iter] [-s stream] [-p prog] [-i infile] [-o outfile]"
  echo "          [-d durafile] [-u usr/pswd] -h"
  echo ""
  echo "-n iter     : number of iterations, default is 100"
  echo "-s stream   : number of streams, default is 2"
  echo "-p prog     : program to run, default is atranspl.ott"
  echo "-i infile   : input file prefix, suffix by process number within a"
  echo "              stream and run ID, default is ./acid_in"
  echo "-o outfile  : output file prefix, similar to input file"
  echo "              default is ./out/acid_out"
  echo "-d durafile : durability file prefix, used for durability tests"
  echo "              default is ./dura/acid_dura"
  echo "-u usr/pswd : user/password combo for database access, default
is tpch/tpch"
  echo "-t trigger  : trigger time between process starts, default is 1
second"
  echo "-h          : print this usage summary"
  exit 1;
}

ITER=600
STEM=${NUM_STREAMS}
let STEM="$STEM + 1" # add one for the update stream
SF=1
PROG=atranspl
IN=${ACID_DIR}/acid_in
DURA_DIR=$ACID_OUT/dura
OUT=$DURA_DIR/drate
DURA=$DURA_DIR/dura
KEY=${DURA_DIR}/key$$_
echo "$$" > ${DURA_DIR}/shellpid
USER=tpch/tpch
TRIG=1
HCNT=duracntb

set -- `getopt "n:s:p:i:o:d:u:ht:f:" "$@"` || usage

# get all the options

while :
do
  case "$1" in
  -n) shift; ITER=$1;;
  -s) shift; STEM=$1;;
  -p) shift; PROG=$1;;
  -i) shift; IN=$1;;
  -o) shift; OUT=$1;;
  -d) shift; DURA=$1;;
  -u) shift; USER=$1;;
  -h) usage; exit 0;;
  -t) shift; TRIG=$1;;
  -f) shift; SF=$1;;
  --) break;;
  esac
  shift;
done


echo "Starting ACID run..."

i=0
T=`expr $STEM \* $TRIG + 6`

# Get history count before the run

sqlplus $USER @cnt_hist > $DURA_DIR/$HCNT 2>&1

while [ $i -lt $STEM ]
do
  randkey $ITER ${SF} u${USER} > ${KEY}${i} &
  i=`expr $i + 1`
done

wait
# perform the consistency

i=0
while [ $i -lt $STEM ]
do
```


© 2004 Hewlett Packard Company. All rights reserved

```
  for j in `head -10 ${KEY}${i} | awk '{printf "%d ",$1}'`
  do
    sqlplus tpch/tpch @consist $j >> $DURA_DIR/duraconsb
  done
  i=`expr $i + 1`
done

echo "Starting Transaction Counting Program"
count_tx.sh $STEM 100 $DURA_DIR &

i=0
while [ $i -lt $STEM ]
do

  $PROG $i $STEM 1 0 i${KEY}${i} o${OUT}${i} d${DURA}${i}
u$USER s1 &
  T=`expr $T - $TRIG`
  i=`expr $i + 1`

done

wait

echo "ACID run completed"
```

sample.sh

```
#!/bin/ksh
#
# $Header: sample.sh 08-aug-99.17:10:00 mpoess Exp $
#
# sample.sh
#
#  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
#
#   NAME
#     sample.sh - <one-line expansion of the name>
#
#   DESCRIPTION
#     <short description of component this file declares/defines>
#
#   NOTES
#     <other useful comments, qualifications, etc.>
#
#   MODIFIED   (MM/DD/YY)
#   mpoess      08/08/99 - Creation
#   mpoess      08/08/99 - Creation
#

# $1 durability output file

. $KIT_DIR/env

cat $1 | grep o_key | awk '{printf "%d \n", $2}' | head -106 >
/tmp/okey$$
cat $1 | grep l_key | awk '{printf "%d \n", $2}' | head -106 > /tmp/lkey$$


paste /tmp/okey$$ /tmp/lkey$$ > /tmp/keys$$
tail -6 /tmp/keys$$ > /tmp/6keys$$

echo "Keys chosen are:"
cat /tmp/6keys$$

i=1
while [ $i -le 6 ]
do

j=`cat /tmp/6keys$$ | tail -${i} | head -1`
sqlplus tpch/tpch @sample $j
i=`expr $i + 1`
done

#/bin/rm -f /tmp/*key*
```

sample.sql

```
Rem
Rem $Header: sample.sql 08-aug-99.17:10:34 mpoess Exp $
Rem
Rem sample.sql
Rem
Rem  Copyright (c) Oracle Corporation 1999. All Rights Reserved.
Rem
Rem    NAME
Rem      sample.sql - <one-line expansion of the name>
Rem
Rem    DESCRIPTION
Rem      <short description of component this file declares/defines>
Rem
Rem    NOTES
Rem      <other useful comments, qualifications, etc.>
Rem
Rem    MODIFIED   (MM/DD/YY)
Rem    mpoess      08/08/99 - Creation
Rem    mpoess      08/08/99 - Created
Rem

alter session set nls_date_format = 'YYYY-MM-DD HH:MI:SS';
select * from history where  h_o_key = &&1 and h_l_key = &&2;

exit;
```


© 2004 Hewlett Packard Company. All rights reserved

Appendix D: Query text and output

1.log

Begin Execution at Fri Jul 9 14:18:57 2004

-- using default substitutions

```
select
l_returnflag,
l_linestatus,
sum(l_quantity) as sum_qty,
sum(l_extendedprice) as sum_base_price,
sum(l_extendedprice * (1 - l_discount)) as sum_disc_price,
sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) as sum_charge,
avg(l_quantity) as avg_qty,
avg(l_extendedprice) as avg_price,
avg(l_discount) as avg_disc,
count(*) as count_order
from
lineitem
where
l_shipdate <= to_date ('1998-12-01','YYYY-MM-DD')  -  90
group by
l_returnflag,
l_linestatus
order by
l_returnflag,
l_linestatus
```

```
L_RETURNFLAG L_LINESTATUS SUM_QTY
SUM_BASE_PRICE
SUM_DISC_PRICE        SUM_CHARGE            AVG_QTY
AVG_PRICE             AVG_DISC              COUNT_ORDER
A          F          37734107.00           56586554400.73
53758257134.87        55909065222.83        25.52
38273.13              0.05                  1478493.00
N          F          991417.00             1487504710.38
1413082168.05         1469649223.19         25.52
38284.47              0.05                  38854.00
N          O          74476040.00           111701729697.74
106118230307.61       110367043872.50       25.50
38249.12              0.05                  2920374.00
R          F          37719753.00           56568041380.90
53741292684.60        55889619119.83        25.51
38250.85              0.05                  1478870.00


4 rows processed.
Query Processed in 6.11 seconds
```

Ended Executing this Stream at Fri Jul 9 14:19:03 2004

Stream Started at 1089400737.76
Stream Ended at 1089400743.87
Stream Processed in 6.11 seconds

SQL statements processed: 1

10.log

Begin Execution at Fri Jul 9 14:20:00 2004

-- using default substitutions

```
select  *  from  (
select
c_custkey,
c_name,
sum(l_extendedprice * (1 - l_discount)) as revenue,
c_acctbal,
n_name,
c_address,
c_phone,
c_comment
from
customer,
orders,
lineitem,
nation
where
c_custkey = o_custkey
and l_orderkey = o_orderkey
and o_orderdate >= to_date ('1993-10-01',  'YYYY-MM-DD')
and o_orderdate < add_months(  to_date( '1993-10-01',  'YYYY-MM-
DD'),  3)
and l_returnflag = 'R'
and c_nationkey = n_nationkey
group by
c_custkey,
c_name,
c_acctbal,
c_phone,
n_name,
c_address,
c_comment
order by
revenue desc)
where rownum <= 20
```

```
C_CUSTKEY             C_NAME                REVENUE
C_ACCTBAL             N_NAME
C_ADDRESS                           C_PHONE
C_COMMENT
57040.00              Customer#000057040    734235.25
632.87                JAPAN
Eioyzjf4pp                          22-895-641-3466
requests sleep blithely about the furiously i
143347.00             Customer#000143347    721002.69
2557.47               EGYPT
1aReFYv,Kw4                         14-742-935-3718
fluffily bold excuses haggle finally after the u
60838.00              Customer#000060838    679127.31
2454.77               BRAZIL
64EaJ5vMAHWJlBOxJklpNc2RJiWE          12-913-494-9813
furiously even pinto beans integrate under the ruthless foxes; ironic,
even dolphins across the slyl
101998.00             Customer#000101998    637029.57
```


© 2004 Hewlett Packard Company. All rights reserved

3790.89 UNITED KINGDOM
01c9CILnNtfOQYmZj 33-593-865-6378
accounts doze blithely! enticing, final deposits sleep blithely special
accounts. slyly express accounts pla
125341.00 Customer#000125341 633508.09
4983.51 GERMANY
S29ODD6bceU8QSuuEJznkNaK 17-582-695-5962
quickly express requests wake quickly blithely
25501.00 Customer#000025501 620269.78
7725.04 ETHIOPIA
W556MXuoiaYCCZamJI,Rn0B4ACUGdkQ8DZ 15-874-808-6793
quickly special requests sleep evenly among the special deposits. special
deposi
115831.00 Customer#000115831 596423.87
5098.10 FRANCE
rFeBbEEyk dl ne7zV5fDrmiq1oK09wV7pxqCgIc 16-715-386-3788
carefully bold excuses sleep alongside of the thinly idle
84223.00 Customer#000084223 594998.02
528.65 UNITED KINGDOM
nAVZCs6BaWap rrM27N 2qBnzc5WBauxbA 33-442-824-8191
pending, final ideas haggle final requests. unusual, regular asymptotes
affix according to the even foxes.
54289.00 Customer#000054289 585603.39
5583.02 IRAN
vXCxoCsU0Bad5JQI ,oobkZ 20-834-292-4707
express requests sublate blithely regular requests. regular, even ideas
solve.
39922.00 Customer#000039922 584878.11
7321.11 GERMANY
Zgy4s50l2GKN4pLDPBU8m342gIw6R 17-147-757-8036
even pinto beans haggle. slyly bold accounts inte
6226.00 Customer#000006226 576783.76
2230.09 UNITED KINGDOM
8gPu8,NPGkfyQQ0hcIYUGPIBWc,ybP5g, 33-657-701-3391
quickly final requests against the regular instructions wake blithely final
instructions. pa
922.00 Customer#000000922 576767.53
3869.25 GERMANY
Az9RFaut7NkPnc5zSD2PwHgVwr4jRzq 17-945-916-9648
boldly final requests cajole blith
147946.00 Customer#000147946 576455.13
2030.13 ALGERIA
iANyZHjqhyy7Ajah0pTrYyhJ 10-886-956-3143
furiously even accounts are blithely above the furiousl
115640.00 Customer#000115640 569341.19
6436.10 ARGENTINA
Vtgfia9qI 7EpHgecU1X 11-411-543-4901
final instructions are slyly according to the
73606.00 Customer#000073606 568656.86
1785.67 JAPAN
xuR0Tro5yChDfOCrjkd2ol 22-437-653-6966
furiously bold orbits about the furiously busy requests wake across the
furiously quiet theodolites. d
110246.00 Customer#000110246 566842.98
7763.35 VIETNAM
7KzflgX MDOq7sOkI 31-943-426-9837
dolphins sleep blithely among the slyly final
142549.00 Customer#000142549 563537.24
5085.99 INDONESIA
ChqEoK43OysjdHbtKCp6dKqjNyvvi9 19-955-562-2398
regular, unusual dependencies boost slyly; ironic attainments nag fluffily
into the unusual packages?
146149.00 Customer#000146149 557254.99
1791.55 ROMANIA
s87fvzFQpU 29-744-164-6487
silent, unusual requests detect quickly slyly regul
52528.00 Customer#000052528 556397.35
551.79 ARGENTINA
NFztyTOR10UOJ 11-208-192-3205
unusual requests detect. slyly dogged theodolites use slyly. deposit
23431.00 Customer#000023431 554269.54
3381.86 ROMANIA
HgiV0phqhaIa9aydNoIlb 29-915-458-2654
instructions nag quickly. furiously bold accounts cajol

20 rows processed.
Query Processed in 5.54 seconds.

Ended Executing this Stream at Fri Jul 9 14:20:05 2004

Stream Started at 1089400800.12
Stream Ended at 1089400805.65
Stream Processed in 5.54 seconds

SQL statements processed: 1

11.log

Begin Execution at Fri Jul 9 14:20:05 2004

-- using default substitutions

```
select
ps_partkey,
sum(ps_supplycost * ps_availqty) as value
from
partsupp,
supplier,
nation
where
ps_suppkey = s_suppkey
and s_nationkey = n_nationkey
and n_name = 'GERMANY'
group by
ps_partkey having
sum(ps_supplycost * ps_availqty) > (
select
sum(ps_supplycost * ps_availqty) * 0.0001000000
from
partsupp,
supplier,
nation
where
ps_suppkey = s_suppkey
and s_nationkey = n_nationkey
and n_name = 'GERMANY'
)
order by
value desc
```

PS_PARTKEY	VALUE
129760.00	17538456.86
166726.00	16503353.92
191287.00	16474801.97
161758.00	16101755.54
34452.00	15983844.72
139035.00	15907078.34
9403.00	15451755.62
154358.00	15212937.88
38823.00	15064802.86
85606.00	15053957.15
33354.00	14408297.40
154747.00	14407580.68

© 2004 Hewlett Packard Company. All rights reserved

```
82865.00        14235489.78
2267.00         7925649.37
45924.00        7925047.68
11493.00        7916722.23
104478.00       7916253.60
166794.00       7913842.00
161995.00       7910874.27
23538.00        7909752.06
41093.00        7909579.92
112073.00       7908617.57
92814.00        7908262.50
88919.00        7907992.50
79753.00        7907933.88
108765.00       7905338.98
146530.00       7905336.60
71475.00        7903367.58
36289.00        7901946.50
61739.00        7900794.00
52338.00        7898638.08
194299.00       7898421.24
105235.00       7897829.94
77207.00        7897752.72
96712.00        7897575.27
10157.00        7897046.25
171154.00       7896814.50
79373.00        7896186.00
113808.00       7893353.88
27901.00        7892952.00
128820.00       7892882.72
25891.00        7890511.20
122819.00       7888881.02
154731.00       7888301.33
101674.00       7879324.60
51968.00        7879102.21
72073.00        7877736.11
5182.00         7874521.73


1048 rows processed.
Query Processed in 2.81 seconds.


Ended Executing this Stream at Fri Jul  9 14:20:08 2004


Stream Started at 1089400805.71
Stream Ended at 1089400808.52
Stream Processed in 2.81 seconds


SQL statements processed: 1
_____________________________________
12.log
_____________________________________
Begin Execution at Fri Jul  9 14:20:08 2004



-- using default substitutions


select
        l_shipmode,
        sum(case
                when o_orderpriority = '1-URGENT'
                        or o_orderpriority = '2-HIGH'
                        then 1
                else 0
        end) as high_line_count,
        sum(case
                when o_orderpriority <> '1-URGENT'
                        and o_orderpriority <> '2-HIGH'
                        then 1
                else 0
        end) as low_line_count
from
        orders,
        lineitem
where
        o_orderkey = l_orderkey
        and l_shipmode in ('MAIL', 'SHIP')
        and l_commitdate < l_receiptdate
        and l_shipdate < l_commitdate
and l_receiptdate >= to_date( '1994-01-01',  'YYYY-MM-DD')
and l_receiptdate < add_months(to_date ('1994-01-01',  'YYYY-MM-
DD'),  12)
group by
        l_shipmode
order by
        l_shipmode

L_SHIPMODE HIGH_LINE_COUNT      LOW_LINE_COUNT
MAIL       6202.00              9324.00
SHIP       6200.00              9262.00


2 rows processed.
Query Processed in 7.10 seconds.


Ended Executing this Stream at Fri Jul  9 14:20:15 2004


Stream Started at 1089400808.58
Stream Ended at 1089400815.68
Stream Processed in 7.10 seconds


SQL statements processed: 1
_____________________________________
13.log
_____________________________________
Begin Execution at Fri Jul  9 14:20:15 2004



-- using default substitutions


select
c_count,
count(*) as custdist
from
(
select
c_custkey,
count(o_orderkey)  as  c_count
from
customer, orders where
c_custkey = o_custkey(+)
and o_comment(+) not like '%special%requests%'
group by
c_custkey
) c_orders
group by
c_count
order by
custdist desc,
c_count desc
```


© 2004 Hewlett Packard Company. All rights reserved

```
C_COUNT          CUSTDIST
0.00             50004.00
9.00             6641.00
10.00            6566.00
11.00            6058.00
8.00             5949.00
12.00            5553.00
13.00            4989.00
19.00            4748.00
7.00             4707.00
18.00            4625.00
15.00            4552.00
17.00            4530.00
14.00            4484.00
20.00            4461.00
16.00            4323.00
21.00            4217.00
22.00            3730.00
6.00             3334.00
23.00            3129.00
24.00            2622.00
25.00            2079.00
5.00             1972.00
26.00            1593.00
27.00            1185.00
4.00             1033.00
28.00            869.00
29.00            559.00
3.00             398.00
30.00            373.00
31.00            235.00
2.00             144.00
32.00            128.00
33.00            71.00
34.00            48.00
35.00            33.00
1.00             23.00
36.00            17.00
37.00            7.00
40.00            4.00
38.00            4.00
39.00            2.00
41.00            1.00

42 rows processed.
Query Processed in 1.99 seconds.


Ended Executing this Stream at Fri Jul  9 14:20:17 2004


Stream Started at 1089400815.74
Stream Ended at 1089400817.73
Stream Processed in 1.99 seconds


SQL statements processed: 1
_____________________________________
14.log
_____________________________________
Begin Execution at Fri Jul  9 14:20:17 2004



-- using default substitutions


select
        100.00 * sum(case
                when p_type like 'PROMO%'
                        then l_extendedprice * (1 - l_discount)
                else 0
        end) / sum(l_extendedprice * (1 - l_discount)) as
promo_revenue
from
        lineitem,
        part
where
        l_partkey = p_partkey
        and l_shipdate >= date '1995-09-01'
        and l_shipdate < date '1995-09-01' + interval '1' month

PROMO_REVENUE
16.38


1 row processed.
Query Processed in 0.39 seconds.


Ended Executing this Stream at Fri Jul  9 14:20:18 2004


Stream Started at 1089400817.79
Stream Ended at 1089400818.17
Stream Processed in 0.39 seconds


SQL statements processed: 1
_____________________________________
15.log
_____________________________________
Begin Execution at Fri Jul  9 14:20:18 2004



-- using default substitutions


create view revenue0 (supplier_no, total_revenue) as
select
l_suppkey,
sum(l_extendedprice * (1 - l_discount))
from
lineitem
where
l_shipdate >= to_date( '1996-01-01', 'YYYY-MM-DD')
and l_shipdate < add_months( to_date ('1996-01-01', 'YYYY-MM-DD'),
3)
group by
l_suppkey
Query Processed in 0.02 seconds.



select
s_suppkey,
s_name,
s_address,
s_phone,
total_revenue
from
supplier,
revenue0
where
s_suppkey = supplier_no
and total_revenue = (
select
```


© 2004 Hewlett Packard Company. All rights reserved

```
max(total_revenue)
from
revenue0
)
order by
s_suppkey

S_SUPPKEY          S_NAME
S_ADDRESS                    S_PHONE       TOTAL_REVENUE
8449.00          Supplier#000008449
Wp34zim9qYFbVctdW                20-469-856-8873 1772627.21


1 row processed.
Query Processed in 0.92 seconds.




drop view revenue0
Query Processed in 0.02 seconds.


Ended Executing this Stream at Fri Jul  9 14:20:19 2004


Stream Started at 1089400818.23
Stream Ended at 1089400819.19
Stream Processed in 0.96 seconds


SQL statements processed: 3
```

16.log

```
Begin Execution at Fri Jul  9 14:20:19 2004



-- using default substitutions


select
p_brand,
p_type,
p_size,
count(distinct ps_suppkey) as supplier_cnt
from
partsupp,
part
where
p_partkey = ps_partkey
and p_brand <> 'Brand#45'
and p_type not like 'MEDIUM POLISHED%'
and p_size in (49, 14, 23, 45, 19, 3, 36, 9)
and ps_suppkey not in (
select
s_suppkey
from
supplier
where
s_comment like '%Customer%Complaints%'
)
group by
p_brand,
p_type,
p_size
order by
supplier_cnt desc,
p_brand,
p_type,
p_size

P_BRAND    P_TYPE                P_SIZE              SUPPLIER_CNT
Brand#41   MEDIUM BRUSHED TIN       3.00                28.00
Brand#54   STANDARD BRUSHED COPPER   14.00               27.00
Brand#11   STANDARD BRUSHED TIN      23.00               24.00
Brand#11   STANDARD BURNISHED BRASS  36.00               24.00
Brand#15   MEDIUM ANODIZED NICKEL    3.00                24.00
Brand#15   SMALL ANODIZED BRASS      45.00               24.00
Brand#15   SMALL BURNISHED NICKEL    19.00               24.00
Brand#21   MEDIUM ANODIZED COPPER    3.00                24.00
Brand#55   SMALL POLISHED BRASS      9.00                4.00
Brand#55   SMALL POLISHED BRASS      19.00               4.00
Brand#55   SMALL POLISHED BRASS      49.00               4.00
Brand#55   SMALL POLISHED COPPER     19.00               4.00
Brand#55   SMALL POLISHED COPPER     23.00               4.00
Brand#55   SMALL POLISHED COPPER     36.00               4.00
Brand#55   SMALL POLISHED COPPER     45.00               4.00
Brand#55   SMALL POLISHED COPPER     49.00               4.00
Brand#55   SMALL POLISHED NICKEL     9.00                4.00
Brand#55   SMALL POLISHED NICKEL     14.00               4.00
Brand#55   SMALL POLISHED NICKEL     19.00               4.00
Brand#55   SMALL POLISHED NICKEL     23.00               4.00
Brand#55   SMALL POLISHED NICKEL     45.00               4.00
Brand#55   SMALL POLISHED NICKEL     49.00               4.00
Brand#55   SMALL POLISHED STEEL      19.00               4.00
Brand#55   SMALL POLISHED STEEL      45.00               4.00
Brand#55   SMALL POLISHED TIN        14.00               4.00
Brand#55   SMALL POLISHED TIN        23.00               4.00
Brand#55   SMALL POLISHED TIN        45.00               4.00
:
:
:
Lines Deleted
:
:
:
Brand#15   LARGE PLATED NICKEL       45.00               3.00
Brand#15   LARGE POLISHED NICKEL     9.00                3.00
Brand#21   PROMO BURNISHED STEEL     45.00               3.00
Brand#22   STANDARD PLATED STEEL     23.00               3.00
Brand#25   LARGE PLATED STEEL        19.00               3.00
Brand#32   STANDARD ANODIZED COPPER  23.00               3.00
Brand#33   SMALL ANODIZED BRASS      9.00                3.00
Brand#35   MEDIUM ANODIZED TIN       19.00               3.00
Brand#51   SMALL PLATED BRASS        23.00               3.00
Brand#52   MEDIUM BRUSHED BRASS      45.00               3.00
Brand#53   MEDIUM BRUSHED TIN        45.00               3.00
Brand#54   ECONOMY POLISHED BRASS    9.00                3.00
Brand#55   PROMO PLATED BRASS        19.00               3.00
Brand#55   STANDARD PLATED TIN       49.00               3.00


18314 rows processed.
Query Processed in 1.58 seconds.


Ended Executing this Stream at Fri Jul  9 14:20:20 2004


Stream Started at 1089400819.24
Stream Ended at 1089400820.82
Stream Processed in 1.58 seconds


SQL statements processed: 1
```

17.log

© 2004 Hewlett Packard Company. All rights reserved

```
Begin Execution at Fri Jul  9 14:20:20 2004


-- using default substitutions


select
sum(l_extendedprice) / 7.0 as avg_yearly
from
lineitem,
part
where
p_partkey = l_partkey
and p_brand = 'Brand#23'
and p_container = 'MED BOX'
and l_quantity < (
select
0.2 * avg(l_quantity)
from
lineitem
where
l_partkey = p_partkey
)

AVG_YEARLY
348406.05


1 row processed.
Query Processed in 5.30 seconds


Ended Executing this Stream at Fri Jul  9 14:20:26 2004


Stream Started at 1089400820.88
Stream Ended at 1089400826.18
Stream Processed in 5.30 seconds


SQL statements processed: 1
_____________________________________
18.log
_____________________________________
Begin Execution at Fri Jul  9 14:20:26 2004


-- using default substitutions


select  *  from  (
select
c_name,
c_custkey,
o_orderkey,
o_orderdate,
o_totalprice,
sum(l_quantity)
from
customer,
orders,
lineitem
where
o_orderkey in (
select
l_orderkey
from
lineitem
group by
l_orderkey having
sum(l_quantity) > 300
)
and c_custkey = o_custkey
and o_orderkey = l_orderkey
group by
c_name,
c_custkey,
o_orderkey,
o_orderdate,
o_totalprice
order by
o_totalprice desc,
o_orderdate
)
where rownum <= 100

C_NAME                C_CUSTKEY          O_ORDERKEY
O_ORDERDATE
O_TOTALPRICE         SUM(L_QUANTITY)
Customer#000128120       128120.00          4722021.00          1994-
04-07
544089.09          323.00
Customer#000144617       144617.00          3043270.00          1997-
02-12
530604.44          317.00
Customer#000013940       13940.00           2232932.00          1997-
04-13
522720.61          304.00
Customer#000066790       66790.00           2199712.00          1996-
09-30
515531.82          327.00
Customer#000046435       46435.00           4745607.00          1997-
07-03
508047.99          309.00
Customer#000015272       15272.00           3883783.00          1993-
07-28
500241.33          302.00
Customer#000146608       146608.00          3342468.00          1994-
06-12
499794.58          303.00
Customer#000096103       96103.00           5984582.00          1992-
03-16
494398.79          312.00
Customer#000024341       24341.00           1474818.00          1992-
11-15
491348.26          302.00
Customer#000137446       137446.00          5489475.00          1997-
05-23
487763.25          311.00
Customer#000107590       107590.00          4267751.00          1994-
11-04
485141.38          301.00
Customer#000050008       50008.00           2366755.00          1996-
12-09
483891.26          302.00
Customer#000015619       15619.00           3767271.00          1996-
08-07
480083.96          318.00
Customer#000077260       77260.00           1436544.00          1992-
09-12
479499.43          307.00
Customer#000109379       109379.00          5746311.00          1996-
10-10
478064.11          302.00
Customer#000054602       54602.00           5832321.00          1997-
02-09
471220.08          307.00
```


© 2004 Hewlett Packard Company. All rights reserved

```
Customer#000105995      105995.00       2096705.00      1994-
07-03
469692.58       307.00
Customer#000148885      148885.00       2942469.00      1992-
05-31
469630.44       313.00
Customer#000114586      114586.00       551136.00       1993-
05-19
469605.59       308.00
Customer#000105260      105260.00       5296167.00      1996-
09-06
469360.57       303.00
Customer#000147197      147197.00       1263015.00      1997-
02-02
467149.67       320.00
Customer#000064483      64483.00        2745894.00      1996-
07-04
466991.35       304.00
Customer#000136573      136573.00       2761378.00      1996-
05-31
461282.73       301.00
Customer#000016384      16384.00        502886.00       1994-
04-12
458378.92       312.00
Customer#000117919      117919.00       2869152.00      1996-
06-20
456815.92       317.00
Customer#000012251      12251.00        735366.00       1993-
11-24
455107.26       309.00
Customer#000120098      120098.00       1971680.00      1995-
06-14
453451.23       308.00
Customer#000066098      66098.00        5007490.00      1992-
08-07
453436.16       304.00
Customer#000117076      117076.00       4290656.00      1997-
02-05
449545.85       301.00
Customer#000129379      129379.00       4720454.00      1997-
06-07
448665.79       303.00
Customer#000126865      126865.00       4702759.00      1994-
11-07
447606.65       320.00
Customer#000088876      88876.00        983201.00       1993-
12-30
446717.46       304.00
Customer#000036619      36619.00        4806726.00      1995-
01-17
446704.09       328.00
Customer#000141823      141823.00       2806245.00      1996-
12-29
446269.12       310.00
Customer#000053029      53029.00        2662214.00      1993-
08-13
446144.49       302.00
Customer#000018188      18188.00        3037414.00      1995-
01-25
443807.22       308.00
Customer#000066533      66533.00        29158.00        1995-
10-21
443576.50       305.00
Customer#000037729      37729.00        4134341.00      1995-
06-29
441082.97       309.00
Customer#000003566      3566.00         2329187.00      1998-
01-04
439803.36       304.00
Customer#000045538      45538.00        4527553.00      1994-
05-22
436275.31       305.00
Customer#000081581      81581.00        4739650.00      1995-
11-04
435405.90       305.00
Customer#000119989      119989.00       1544643.00      1997-
09-20
434568.25       320.00
Customer#000003680      3680.00         3861123.00      1998-
07-03
433525.97       301.00
Customer#000113131      113131.00       967334.00       1995-
12-15
432957.75       301.00
Customer#000141098      141098.00       565574.00       1995-
09-24
430986.69       301.00
Customer#000093392      93392.00        5200102.00      1997-
01-22
425487.51       304.00
Customer#000015631      15631.00        1845057.00      1994-
05-12
419879.59       302.00
Customer#000112987      112987.00       4439686.00      1996-
09-17
418161.49       305.00
Customer#000012599      12599.00        4259524.00      1998-
02-12
415200.61       304.00
Customer#000105410      105410.00       4478371.00      1996-
03-05
412754.51       302.00
Customer#000149842      149842.00       5156581.00      1994-
05-30
411329.35       302.00
Customer#000010129      10129.00        5849444.00      1994-
03-21
409129.85       309.00
Customer#000069904      69904.00        1742403.00      1996-
10-19
408513.00       305.00
Customer#000017746      17746.00        6882.00         1997-
04-09
408446.93       303.00
Customer#000013072      13072.00        1481925.00      1998-
03-15
399195.47       301.00
Customer#000082441      82441.00        857959.00       1994-
02-07
382579.74       305.00
Customer#000088703      88703.00        2995076.00      1994-
01-30
363812.12       302.00

57 rows processed.
Query Processed in 6.27 seconds.

Ended Executing this Stream at Fri Jul  9 14:20:32 2004

Stream Started at 1089400826.24
Stream Ended at 1089400832.52
Stream Processed in 6.27 seconds

SQL statements processed: 1
```


© 2004 Hewlett Packard Company. All rights reserved

19.log

Begin Execution at Fri Jul 9 14:20:32 2004

```
-- using default substitutions

select
sum(l_extendedprice* (1 - l_discount)) as revenue
from
lineitem,
part
where
(
p_partkey = l_partkey
and p_brand = 'Brand#12'
and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
and l_quantity >= 1 and l_quantity <= 1 + 10
and p_size between 1 and 5
and l_shipmode in ('AIR', 'AIR REG')
and l_shipinstruct = 'DELIVER IN PERSON'
)
or
(
p_partkey = l_partkey
and p_brand = 'Brand#23'
and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED
PACK')
and l_quantity >= 10 and l_quantity <= 10 + 10
and p_size between 1 and 10
and l_shipmode in ('AIR', 'AIR REG')
and l_shipinstruct = 'DELIVER IN PERSON'
)
or
(
p_partkey = l_partkey
and p_brand = 'Brand#34'
and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
and l_quantity >= 20 and l_quantity <= 20 + 10
and p_size between 1 and 15
and l_shipmode in ('AIR', 'AIR REG')
and l_shipinstruct = 'DELIVER IN PERSON'
)

REVENUE
3083843.06


1 row processed.
Query Processed in 4.67 seconds


Ended Executing this Stream at Fri Jul  9 14:20:37 2004


Stream Started at 1089400832.57
Stream Ended at 1089400837.24
Stream Processed in 4.67 seconds


SQL statements processed: 1
```

2.log

Begin Execution at Fri Jul 9 14:19:03 2004

```
-- using default substitutions


select * from (
select
s_acctbal,
s_name,
n_name,
p_partkey,
p_mfgr,
s_address,
s_phone,
s_comment
from
part,
supplier,
partsupp,
nation,
region
where
p_partkey = ps_partkey
and s_suppkey = ps_suppkey
and p_size = 15
and p_type like '%BRASS'
and s_nationkey = n_nationkey
and n_regionkey = r_regionkey
and r_name = 'EUROPE'
and ps_supplycost = (
select
min(ps_supplycost)
from
partsupp,
supplier,
nation,
region
where
p_partkey = ps_partkey
and s_suppkey = ps_suppkey
and s_nationkey = n_nationkey
and n_regionkey = r_regionkey
and r_name = 'EUROPE'
)
order by
s_acctbal desc,
n_name,
s_name,
p_partkey
)
where rownum <= 100

S_ACCTBAL          S_NAME                N_NAME
P_PARTKEY          P_MFGR
S_ADDRESS                        S_PHONE
S_COMMENT
9938.53            Supplier#000005359    UNITED KINGDOM
185358.00          Manufacturer#4
QKuHYh,vZGiwu2FWEJoLDx04              33-429-790-6131
blithely silent pinto beans are furiously. slyly final deposits acros
9937.84            Supplier#000005969    ROMANIA
108438.00          Manufacturer#1
ANDENSOSmk,miq23Xfb5RWt6dvUcvt6Qa     29-520-692-3537
carefully slow deposits use furiously. slyly ironic platelets above the
ironic
9936.22            Supplier#000005250    UNITED KINGDOM
249.00             Manufacturer#4
B3rqp0xbSEim4Mpy2RH J                 33-320-228-2957
blithely special packages are. stealthily express deposits across the
closely final instructi
9923.77            Supplier#000002324    GERMANY
29821.00           Manufacturer#4
```


© 2004 Hewlett Packard Company. All rights reserved

```
y3OD9UywSTOk                    17-779-299-1839
quickly express packages breach quiet pinto beans. requ
:
:
:
Lines Deleted
:
:
:

WCNfBPZeSXh3h,c                 32-454-883-3821
blithely regular deposits
7850.66         Supplier#000001518      UNITED KINGDOM
86501.00        Manufacturer#1
ONda3YJiHKJOC                   33-730-383-3892
furiously final accounts wake carefully idle requests. even dolphins
wake acc
7843.52         Supplier#000006683      FRANCE
11680.00        Manufacturer#4
2Z0JGkiv01Y00oCFwUGfviIbhzCdy           16-464-517-8943
carefully bold accounts doub


100 rows processed.
Query Processed in 5.76 seconds


Ended Executing this Stream at Fri Jul  9 14:19:09 2004


Stream Started at 1089400743.93
Stream Ended at 1089400749.70
Stream Processed in 5.76 seconds


SQL statements processed: 1
```

20.log

Begin Execution at Fri Jul 9 14:20:37 2004

```
-- using default substitutions


select  /* use_hash (Supplier) */
s_name,
s_address
from
supplier,
nation
where
s_suppkey in (
select /* use_hash (partsupp part) */
ps_suppkey
from
partsupp
where
ps_partkey in (
select
p_partkey
from
part
where
p_name like 'forest%'
)
and ps_availqty > (
select
0.5 * sum(l_quantity)
from
lineitem
where
l_partkey = ps_partkey
and l_suppkey = ps_suppkey
and l_shipdate >= to_date ('1994-01-01', 'YYYY-MM-DD')
and l_shipdate < add_months( to_date ('1994-01-01', 'YYYY-MM-DD'),
12)
)
)
and s_nationkey = n_nationkey
and n_name = 'CANADA'
order by
s_name

S_NAME                  S_ADDRESS
Supplier#000000020      iybAE,RmTymrZVYaFZva2SH,j
Supplier#000000091
YV45D7TkfdQanOOZ7q9QxkyGUapU1oOWU6q3
Supplier#000000197      YC2Acon6kjY3zj3Fbxs2k4Vdf7X0cd2F
Supplier#000000226      83qOdU2EYRdPQAQhEtn GRZEd
Supplier#000000285      Br7e1nnt1yxrw6ImgpJ7YdhFDjuBf
Supplier#000000378      FfbhyCxWvcPrO8ltp9
Supplier#000000402      i9Sw4DoyMhzhKXCH9By,AYSgmD

:
:
:
Lines Deleted
:
:
:

Supplier#000009278      RqYTzgxj93CLX 0mcYfCENOefD
Supplier#000009327      uoqMdf7e7Gj9dbQ53
Supplier#000009430      igRqmneFt
Supplier#000009567      r4Wfx4c3xsEAjcGj71HHZByornl D9vrztXlv4
Supplier#000009601      51m637bO,Rw5DnHWFUvLacRx9
Supplier#000009709      rRnCbHYgDgl9PZYnyWKVYSUW0vKg
Supplier#000009753      wLhVEcRmd7PkJF4FBnGK7Z
Supplier#000009796      z,y4Idmr15DOvPUqYG
Supplier#000009799       4wNjXGa4OKWl
Supplier#000009811      E3iuyq7UnZxU7oPZIe2Gu6
Supplier#000009812      APFRMy3lCbgFga53n5t9DxzFPQPgnjrGt32
Supplier#000009862      rJzweWeN58
Supplier#000009868      ROjGgx5gvtkmnUUoeyy7v
Supplier#000009869
ucLqxzrpBTRMewGSM29t0rNTM30g1Tu3Xgg3mKag
Supplier#000009899      7XdpAHrzr1t,UQFZE
Supplier#000009974      7wJ,J5DKcxSU4Kp1cQLpbcAvB5AsvKT


204 rows processed.
Query Processed in 2.44 seconds


Ended Executing this Stream at Fri Jul  9 14:20:39 2004


Stream Started at 1089400837.29
Stream Ended at 1089400839.73
Stream Processed in 2.44 seconds


SQL statements processed: 1
```

21.log

Begin Execution at Fri Jul 9 14:20:39 2004

© 2004 Hewlett Packard Company. All rights reserved

```
-- using default substitutions


select * from (
select
s_name,
count(*)  numwait
from
supplier,
lineitem l1,
orders,
nation
where
s_suppkey = l1.l_suppkey
and o_orderkey = l1.l_orderkey
and o_orderstatus = 'F'
and l1.l_receiptdate > l1.l_commitdate
and exists (
select
*
from
lineitem l2
where
l2.l_orderkey = l1.l_orderkey
and l2.l_suppkey <> l1.l_suppkey
)
and not exists (
select
*
from
lineitem l3
where
l3.l_orderkey = l1.l_orderkey
and l3.l_suppkey <> l1.l_suppkey
and l3.l_receiptdate > l3.l_commitdate
)
and s_nationkey = n_nationkey
and n_name = 'SAUDI ARABIA'
group by
s_name
order by
numwait desc,
s_name)
where rownum <= 100
```

S_NAME	NUMWAIT
Supplier#000002829	20.00
Supplier#000005808	18.00
Supplier#000000262	17.00
Supplier#000000496	17.00
Supplier#000002160	17.00
Supplier#000002301	17.00
Supplier#000002540	17.00
Supplier#000003063	17.00
Supplier#000005178	17.00
Supplier#000008331	17.00
Supplier#000002005	16.00
Supplier#000002095	16.00
Supplier#000005799	16.00
Supplier#000005842	16.00
Supplier#000006450	16.00
Supplier#000006939	16.00
Supplier#000009200	16.00
Supplier#000009727	16.00
Supplier#000000486	15.00
Supplier#000000565	15.00
Supplier#000001046	15.00
Supplier#000001047	15.00
Supplier#000001161	15.00
Supplier#000001336	15.00
Supplier#000001435	15.00
Supplier#000003075	15.00
Supplier#000003335	15.00
Supplier#000005649	15.00
Supplier#000006027	15.00
Supplier#000006795	15.00
Supplier#000006800	15.00
Supplier#000006824	15.00
Supplier#000007131	15.00
Supplier#000007382	15.00
Supplier#000008913	15.00
Supplier#000009787	15.00
Supplier#000000633	14.00
Supplier#000001960	14.00
Supplier#000002323	14.00
Supplier#000002490	14.00
Supplier#000002993	14.00
Supplier#000003101	14.00
Supplier#000004489	14.00
Supplier#000005435	14.00
Supplier#000005583	14.00
Supplier#000005774	14.00
Supplier#000007579	14.00
Supplier#000008180	14.00
Supplier#000008695	14.00
Supplier#000009224	14.00
Supplier#000000357	13.00
Supplier#000000436	13.00
Supplier#000000610	13.00
Supplier#000000788	13.00
Supplier#000000889	13.00
Supplier#000001062	13.00
Supplier#000001498	13.00
Supplier#000002056	13.00
Supplier#000002312	13.00
Supplier#000002344	13.00
Supplier#000002596	13.00
Supplier#000002615	13.00
Supplier#000002978	13.00
Supplier#000003048	13.00
Supplier#000003234	13.00
Supplier#000003727	13.00
Supplier#000003806	13.00
Supplier#000004472	13.00
Supplier#000005236	13.00
Supplier#000005906	13.00
Supplier#000006241	13.00
Supplier#000006326	13.00
Supplier#000006384	13.00
Supplier#000006394	13.00
Supplier#000006624	13.00
Supplier#000006629	13.00
Supplier#000006682	13.00
Supplier#000006737	13.00
Supplier#000006825	13.00
Supplier#000007021	13.00
Supplier#000007417	13.00
Supplier#000007497	13.00
Supplier#000007602	13.00
Supplier#000008134	13.00
Supplier#000008234	13.00
Supplier#000009435	13.00
Supplier#000009436	13.00
Supplier#000009564	13.00
Supplier#000009896	13.00
Supplier#000000379	12.00
Supplier#000000673	12.00
Supplier#000000762	12.00

© 2004 Hewlett Packard Company. All rights reserved

```
Supplier#000000811      12.00
Supplier#000000821      12.00
Supplier#000001337      12.00
Supplier#000001916      12.00
Supplier#000001925      12.00
Supplier#000002039      12.00
Supplier#000002357      12.00
Supplier#000002483      12.00


100 rows processed.
Query Processed in 19.29 seconds.


Ended Executing this Stream at Fri Jul  9 14:20:59 2004


Stream Started at 1089400839.79
Stream Ended at 1089400859.08
Stream Processed in 19.29 seconds


SQL statements processed: 1
_____________________________________
22.log
_____________________________________
Begin Execution at Fri Jul  9 14:20:59 2004


-- using default substitutions


select
cntrycode,
count(*) as numcust,
sum(c_acctbal) as totacctbal
from
(
select
substr(c_phone, 1, 2) as cntrycode,
c_acctbal
from
customer
where
substr(c_phone,1, 2) in
('13', '31', '23', '29', '30', '18', '17')
and c_acctbal > (
select
avg(c_acctbal)
from
customer
where
c_acctbal > 0.00
and substr(c_phone, 1, 2) in
('13', '31', '23', '29', '30', '18', '17')
)
and not exists (
select
*
from
orders
where
o_custkey = c_custkey
)
) custsale
group by
cntrycode
order by
cntrycode

CNTRYCODE NUMCUST          TOTACCTBAL
13        888.00           6737713.99
17        861.00           6460573.72
18        964.00           7236687.40
23        892.00           6701457.95
29        948.00           7158866.63
30        909.00           6808436.13
31        922.00           6806670.18


7 rows processed.
Query Processed in 5.34 seconds.


Ended Executing this Stream at Fri Jul  9 14:21:04 2004


Stream Started at 1089400859.14
Stream Ended at 1089400864.48
Stream Processed in 5.34 seconds


SQL statements processed: 1
_____________________________________
3.log
_____________________________________
Begin Execution at Fri Jul  9 14:19:09 2004


-- using default substitutions


select * from (
select
l_orderkey,
sum(l_extendedprice * (1 - l_discount)) as revenue,
o_orderdate,
o_shippriority
from
customer,
orders,
lineitem
where
c_mktsegment = 'BUILDING'
and c_custkey = o_custkey
and l_orderkey = o_orderkey
and o_orderdate < to_date( '1995-03-15', 'YYYY-MM-DD')
and l_shipdate > to_date( '1995-03-15', 'YYYY-MM-DD')
group by
l_orderkey,
o_orderdate,
o_shippriority
order by
revenue desc,
o_orderdate)
where rownum <= 10

L_ORDERKEY          REVENUE          O_ORDERDATE
O_SHIPPRIORITY
2456423.00          406181.01        1995-03-05  0.00
3459808.00          405838.70        1995-03-04  0.00
492164.00           390324.06        1995-02-19  0.00
1188320.00          384537.94        1995-03-09  0.00
2435712.00          378673.06        1995-02-26  0.00
4878020.00          378376.80        1995-03-12  0.00
5521732.00          375153.92        1995-03-13  0.00
2628192.00          373133.31        1995-02-22  0.00
993600.00           371407.46        1995-03-05  0.00
```


© 2004 Hewlett Packard Company. All rights reserved

```
2300070.00          367371.15          1995-03-13  0.00

10 rows processed.
Query Processed in 5.48 seconds.

Ended Executing this Stream at Fri Jul  9 14:19:15 2004

Stream Started at 1089400749.75
Stream Ended at 1089400755.23
Stream Processed in 5.48 seconds

SQL statements processed: 1
```

4.log

Begin Execution at Fri Jul 9 14:19:15 2004

```
-- using default substitutions

select
o_orderpriority,
count(*) as order_count
from
orders
where
o_orderdate >= to_date( '1993-07-01',  'YYYY-MM-DD')
and o_orderdate < add_months(to_date( '1993-07-01', 'YYYY-MM-
DD'),3)
and exists (
select
*
from
lineitem
where
l_orderkey = o_orderkey
and l_commitdate < l_receiptdate
)
group by
o_orderpriority
order by
o_orderpriority

O_ORDERPRIORITY ORDER_COUNT
1-URGENT        10594.00
2-HIGH          10476.00
3-MEDIUM        10410.00
4-NOT SPECIFIED 10556.00
5-LOW           10487.00

5 rows processed.
Query Processed in 4.95 seconds.

Ended Executing this Stream at Fri Jul  9 14:19:20 2004

Stream Started at 1089400755.30
Stream Ended at 1089400760.24
Stream Processed in 4.95 seconds

SQL statements processed: 1
```

5.log

Begin Execution at Fri Jul 9 14:19:20 2004

```
-- using default substitutions

select
n_name,
sum(l_extendedprice * (1 - l_discount)) as revenue
from
customer,
orders,
lineitem,
supplier,
nation,
region
where
c_custkey = o_custkey
and l_orderkey = o_orderkey
and l_suppkey = s_suppkey
and c_nationkey = s_nationkey
and s_nationkey = n_nationkey
and n_regionkey = r_regionkey
and r_name = 'ASIA'
and o_orderdate >= to_date( '1994-01-01',  'YYYY-MM-DD')
and o_orderdate < add_months(to_date( '1994-01-01',  'YYYY-MM-
DD'),  12)
group by
n_name
order by
revenue desc

N_NAME                REVENUE
INDONESIA           55502041.17
VIETNAM             55295087.00
CHINA               53724494.26
INDIA               52035512.00
JAPAN               45410175.70

5 rows processed.
Query Processed in 6.06 seconds.

Ended Executing this Stream at Fri Jul  9 14:19:26 2004

Stream Started at 1089400760.30
Stream Ended at 1089400766.37
Stream Processed in 6.06 seconds

SQL statements processed: 1
```

6.log

Begin Execution at Fri Jul 9 14:19:26 2004

```
-- using default substitutions

select
sum(l_extendedprice * l_discount) as revenue
from
```


© 2004 Hewlett Packard Company. All rights reserved

```
lineitem
where
l_shipdate >= to_date( '1994-01-01',  'YYYY-MM-DD')
and l_shipdate < add_months(to_date( '1994-01-01',  'YYYY-MM-DD'),
12)
and l_discount between .06 - 0.01 and .06 + 0.01
and l_quantity < 24

REVENUE
123141078.23


1 row processed.
Query Processed in 0.85 seconds.


Ended Executing this Stream at Fri Jul  9 14:19:27 2004


Stream Started at 1089400766.43
Stream Ended at 1089400767.27
Stream Processed in 0.85 seconds


SQL statements processed: 1
```

7.log

```
Begin Execution at Fri Jul  9 14:19:27 2004



-- using default substitutions


select
supp_nation,
cust_nation,
l_year,
sum(volume) as revenue
from
(
select
n1.n_name as supp_nation,
n2.n_name as cust_nation,
                                        to_number (to_char
(l_shipdate,'yyyy')) as l_year,
l_extendedprice * (1 - l_discount) as volume
from
supplier,
lineitem,
orders,
customer,
nation n1,
nation n2
where
s_suppkey = l_suppkey
and o_orderkey = l_orderkey
and c_custkey = o_custkey
and s_nationkey = n1.n_nationkey
and c_nationkey = n2.n_nationkey
and (
(n1.n_name = 'FRANCE' and n2.n_name = 'GERMANY')
or (n1.n_name = 'GERMANY' and n2.n_name = 'FRANCE')
)
and l_shipdate between to_date( '1995-01-01', 'YYYY-MM-DD') and
to_date( '1996-12-31', 'YYYY-MM-DD')
) shipping
group by
supp_nation,
cust_nation,
l_year
order by
supp_nation,
cust_nation,
l_year

SUPP_NATION          CUST_NATION          L_YEAR
REVENUE
FRANCE               GERMANY              1995.00
54639732.73
FRANCE               GERMANY              1996.00
54633083.31
GERMANY                FRANCE             1995.00
52531746.67
GERMANY                FRANCE             1996.00
52520549.02


4 rows processed.
Query Processed in 6.81 seconds.


Ended Executing this Stream at Fri Jul  9 14:19:34 2004


Stream Started at 1089400767.33
Stream Ended at 1089400774.14
Stream Processed in 6.81 seconds


SQL statements processed: 1
```

8.log

```
Begin Execution at Fri Jul  9 14:19:34 2004



-- using default substitutions


select
o_year,
sum(case when nation='BRAZIL' then volume else 0 end )/
sum(volume)
as mkt_share
from
(
select
to_number (to_char (o_orderdate, 'yyyy')) as o_year,
l_extendedprice * (1 - l_discount) as volume,
n2.n_name as nation
from
part,
supplier,
lineitem,
orders,
customer,
nation n1,
nation n2,
region
where
p_partkey = l_partkey
and s_suppkey = l_suppkey
and l_orderkey = o_orderkey
and o_custkey = c_custkey
and c_nationkey = n1.n_nationkey
and n1.n_regionkey = r_regionkey
and r_name = 'AMERICA'
```


© 2004 Hewlett Packard Company. All rights reserved

```
and s_nationkey = n2.n_nationkey
and o_orderdate between to_date ('1995-01-01', 'YYYY-MM-DD') and
to_date ('1996-12-31', 'YYYY-MM-DD')
and p_type = 'ECONOMY ANODIZED STEEL'
) all_nations
group by
o_year
order by
o_year
```

O_YEAR	MKT_SHARE
1995.00	0.03
1996.00	0.04

2 rows processed.
Query Processed in 11.80 seconds.

Ended Executing this Stream at Fri Jul 9 14:19:46 2004

Stream Started at 1089400774.20
Stream Ended at 1089400786.00
Stream Processed in 11.80 seconds

SQL statements processed: 1

9.log

Begin Execution at Fri Jul 9 14:19:46 2004

```
-- using default substitutions

select
nation,
o_year,
sum(amount) as sum_profit
from
(
select
n_name as nation,
to_number (to_char (o_orderdate, 'yyyy')) as o_year,
l_extendedprice * (1 - l_discount) - ps_supplycost * l_quantity as
amount
from
part,
supplier,
                                        lineitem,
partsupp,
orders,
                                        nation
where
s_suppkey = l_suppkey
and ps_suppkey = l_suppkey
and ps_partkey = l_partkey
and p_partkey = l_partkey
and o_orderkey = l_orderkey
and s_nationkey = n_nationkey
and p_name like '%green%'
) profit
group by
nation,
o_year
order by
nation,
o_year desc
```

NATION	O_YEAR	SUM_PROFIT
ALGERIA	1998.00	31342867.23
ALGERIA	1997.00	57138193.02
ALGERIA	1996.00	56140140.13
ALGERIA	1995.00	53051469.65
ALGERIA	1994.00	53867582.13
ALGERIA	1993.00	54942718.13
ALGERIA	1992.00	54628034.71
ARGENTINA	1998.00	30211185.71
ARGENTINA	1997.00	50805741.75
ARGENTINA	1996.00	51923746.58
ARGENTINA	1995.00	49298625.77
ARGENTINA	1994.00	50835610.11
ARGENTINA	1993.00	51646079.18
ARGENTINA	1992.00	50410314.99
BRAZIL	1998.00	27217924.38
BRAZIL	1997.00	48378669.20
BRAZIL	1996.00	50482870.36
BRAZIL	1995.00	47623383.63
BRAZIL	1994.00	47840165.73
BRAZIL	1993.00	49054694.04
BRAZIL	1992.00	48667639.08
CANADA	1998.00	30379833.77
CANADA	1997.00	50465052.31
CANADA	1996.00	52560501.39
CANADA	1995.00	52375332.81
CANADA	1994.00	52600364.66
CANADA	1993.00	52644504.07
CANADA	1992.00	53932871.70
CHINA	1998.00	31075466.16
CHINA	1997.00	50551874.45
CHINA	1996.00	51039293.88
CHINA	1995.00	49287534.62
CHINA	1994.00	50851090.07
CHINA	1993.00	54229629.83
CHINA	1992.00	52400529.37
EGYPT	1998.00	29054433.39
EGYPT	1997.00	50627611.45
EGYPT	1996.00	49542212.84
EGYPT	1995.00	48311550.32
EGYPT	1994.00	49790644.74
EGYPT	1993.00	48904292.97
EGYPT	1992.00	49434932.62
ETHIOPIA	1998.00	28040717.27
ETHIOPIA	1997.00	47455009.87
ETHIOPIA	1996.00	46491097.57
ETHIOPIA	1995.00	46804449.30
ETHIOPIA	1994.00	48516143.92
ETHIOPIA	1993.00	46551891.56
ETHIOPIA	1992.00	44934648.64
FRANCE	1998.00	32226407.84
FRANCE	1997.00	47121485.86
FRANCE	1996.00	47263135.50
FRANCE	1995.00	47275997.57
FRANCE	1994.00	47067209.33
FRANCE	1993.00	51163370.11
FRANCE	1992.00	47846235.33
GERMANY	1998.00	28624942.66
GERMANY	1997.00	49309074.88
GERMANY	1996.00	49918683.17
GERMANY	1995.00	52650718.72
GERMANY	1994.00	50346900.42
GERMANY	1993.00	50991895.81
GERMANY	1992.00	48274126.10
INDIA	1998.00	29943144.35
INDIA	1997.00	50665453.23
INDIA	1996.00	50283092.29
INDIA	1995.00	50006774.64

© 2004 Hewlett Packard Company. All rights reserved

INDIA	1994.00	48995190.76
INDIA	1993.00	50286902.85
INDIA	1992.00	50850329.40
INDONESIA	1998.00	27672340.00
INDONESIA	1997.00	50512145.73
INDONESIA	1996.00	51653060.12
INDONESIA	1995.00	51508779.59
INDONESIA	1994.00	52817950.32
INDONESIA	1993.00	47959994.96
INDONESIA	1992.00	51776605.03
IRAN	1998.00	29065736.24
IRAN	1997.00	50042063.05
IRAN	1996.00	50926653.19
IRAN	1995.00	51249667.65
IRAN	1994.00	50337085.87
IRAN	1993.00	51730763.49
IRAN	1992.00	49955856.56
IRAQ	1998.00	31624551.00
IRAQ	1997.00	55121749.02
IRAQ	1996.00	55897663.79
IRAQ	1995.00	54815472.52
IRAQ	1994.00	54408516.13
IRAQ	1993.00	53633167.98
IRAQ	1992.00	55891939.34
JAPAN	1998.00	27934179.67
JAPAN	1997.00	44517162.55
JAPAN	1996.00	42545606.12
JAPAN	1995.00	43749356.40
JAPAN	1994.00	44840243.07
JAPAN	1993.00	44660015.53
JAPAN	1992.00	45410249.12
JORDAN	1998.00	26901488.58
JORDAN	1997.00	45471878.41
JORDAN	1996.00	46794325.79
JORDAN	1995.00	45178828.58
JORDAN	1994.00	45333636.51
JORDAN	1993.00	47971496.10
JORDAN	1992.00	44717239.18
KENYA	1998.00	28597614.34
KENYA	1997.00	47949733.73
KENYA	1996.00	46886924.62
KENYA	1995.00	46072338.76
KENYA	1994.00	45772061.17
KENYA	1993.00	46308728.23
KENYA	1992.00	47257780.84
MOROCCO	1998.00	26732115.58
MOROCCO	1997.00	45637304.25
MOROCCO	1996.00	45558221.75
MOROCCO	1995.00	47851318.89
MOROCCO	1994.00	46272172.94
MOROCCO	1993.00	46764326.18
MOROCCO	1992.00	48122783.58
MOZAMBIQUE	1998.00	30712392.01
MOZAMBIQUE	1997.00	50316528.76
MOZAMBIQUE	1996.00	51640320.25
MOZAMBIQUE	1995.00	50693774.51
MOZAMBIQUE	1994.00	49253277.63
MOZAMBIQUE	1993.00	49153016.54
MOZAMBIQUE	1992.00	48247551.85
PERU	1998.00	29326102.32
PERU	1997.00	49753780.40
PERU	1996.00	50935170.29
PERU	1995.00	53309883.41
PERU	1994.00	50643531.80
PERU	1993.00	51584622.00
PERU	1992.00	47523899.05
ROMANIA	1998.00	30368667.40
ROMANIA	1997.00	50365683.85
ROMANIA	1996.00	49598999.01
ROMANIA	1995.00	47537642.87
ROMANIA	1994.00	51455283.01
ROMANIA	1993.00	50407136.89
ROMANIA	1992.00	48185385.13
RUSSIA	1998.00	28322384.03
RUSSIA	1997.00	50106685.18
RUSSIA	1996.00	51753342.43
RUSSIA	1995.00	49215820.36
RUSSIA	1994.00	52205666.44
RUSSIA	1993.00	51860230.03
RUSSIA	1992.00	53251677.15
SAUDI ARABIA	1998.00	31541259.81
SAUDI ARABIA	1997.00	52438750.81
SAUDI ARABIA	1996.00	52543737.82
SAUDI ARABIA	1995.00	52938696.53
SAUDI ARABIA	1994.00	51389601.97
SAUDI ARABIA	1993.00	52937508.88
SAUDI ARABIA	1992.00	54843459.64
UNITED KINGDOM	1998.00	28494874.00
UNITED KINGDOM	1997.00	49381810.90
UNITED KINGDOM	1996.00	51386853.96
UNITED KINGDOM	1995.00	51509586.79
UNITED KINGDOM	1994.00	48086499.71
UNITED KINGDOM	1993.00	49166827.22
UNITED KINGDOM	1992.00	49349122.08
UNITED STATES	1998.00	25126238.95
UNITED STATES	1997.00	50077306.42
UNITED STATES	1996.00	48048649.47
UNITED STATES	1995.00	48809032.42
UNITED STATES	1994.00	49296747.18
UNITED STATES	1993.00	48029946.80
UNITED STATES	1992.00	48671944.50
VIETNAM	1998.00	30442736.06
VIETNAM	1997.00	50309179.79
VIETNAM	1996.00	50488161.41
VIETNAM	1995.00	49658284.61
VIETNAM	1994.00	50596057.26
VIETNAM	1993.00	50953919.15
VIETNAM	1992.00	49613838.32

175 rows processed.
Query Processed in 14.00 seconds.

Ended Executing this Stream at Fri Jul 9 14:20:00 2004

Stream Started at 1089400786.06
Stream Ended at 1089400800.06
Stream Processed in 14.00 seconds

SQL statements processed: 1

© 2004 Hewlett Packard Company. All rights reserved

Appendix E: Seed and input parameters

Seed

117143112

qp1.0

14 1994-08-01
2 45 TIN AFRICA
9 moccasin
20 dodger 1996-01-01 SAUDI ARABIA
6 1996-01-01 0.06 25
17 Brand#33 JUMBO PACK
18 314
8 UNITED KINGDOM EUROPE LARGE PLATED
COPPER
21 ROMANIA
13 unusual requests
3 HOUSEHOLD 1995-03-06
22 17 22 10 12 11 29
23
16 Brand#33 LARGE ANODIZED 9 14 39
42 26 34 47 25
4 1993-11-01
11 FRANCE 0.0000000333
15 1993-06-01
1 111
10 1994-09-01
19 Brand#42 Brand#45 Brand#35 9 15 22
5 ASIA 1996-01-01
7 KENYA UNITED KINGDOM
12 MAIL RAIL 1993-01-01

qp1.1

21 IRAQ
3 BUILDING 1995-03-22
18 315
5 EUROPE 1996-01-01
11 ROMANIA 0.0000000333
7 ETHIOPIA MOROCCO
6 1996-01-01 0.03 25
20 peach 1995-01-01 IRAN
17 Brand#35 JUMBO DRUM
12 TRUCK MAIL 1994-01-01
16 Brand#13 PROMO PLATED 40 26 14
25 48 34 33 28
15 1996-01-01
13 unusual requests
10 1993-06-01
2 33 COPPER EUROPE
8 MOROCCO AFRICA LARGE ANODIZED
COPPER
14 1994-11-01
19 Brand#44 Brand#33 Brand#34 4 16 29
9 maroon
22 11 23 28 30 10 29
14
1 119
4 1996-06-01

qp1.2

6 1997-01-01 0.09 24
17 Brand#32 WRAP BOX
14 1995-02-01
16 Brand#53 SMALL POLISHED 29 5 31
43 10 16 41 32
19 Brand#51 Brand#11 Brand#23 9 17 26
10 1994-03-01
9 lawn
2 20 STEEL AMERICA
15 1993-09-01
8 GERMANY EUROPE MEDIUM POLISHED
COPPER
5 MIDDLE EAST 1997-01-01
22 10 19 12 31 25 23
22
12 AIR MAIL 1995-01-01
7 RUSSIA GERMANY
13 unusual requests
18 313
1 66
4 1994-03-01
20 blush 1993-01-01 ALGERIA
3 HOUSEHOLD 1995-03-08
11 GERMANY 0.0000000333
21 CANADA

qp1.3

8 UNITED STATES AMERICA MEDIUM
BURNISHED COPPER
5 AFRICA 1997-01-01
4 1996-10-01
6 1997-01-01 0.06 25
17 Brand#34 WRAP PACK
7 KENYA UNITED STATES
1 74
18 314
22 25 17 19 12 14 16
22
14 1995-05-01
9 hot
10 1995-01-01
15 1996-04-01
11 SAUDI ARABIA 0.0000000333
20 magenta 1997-01-01 MOROCCO
2 8 BRASS EUROPE
21 SAUDI ARABIA
19 Brand#53 Brand#54 Brand#22 4 18 22
13 unusual requests

© 2004 Hewlett Packard Company. All rights reserved

```
16      Brand#33  ECONOMY ANODIZED        46      26
        42      31      33      19      12      39
12      REG AIR  FOB      1995-01-01
3       AUTOMOBILE      1995-03-24
```

qp1.4

```
5       AMERICA         1997-01-01
21      JAPAN
14      1995-08-01
19      Brand#55  Brand#31  Brand#22  10      19      29
15      1994-01-01
17      Brand#31  WRAP DRUM
12      SHIP      FOB      1995-01-01
6       1997-01-01      0.04      25
4       1994-07-01
9       gainsboro
8       MOZAMBIQUE      AFRICA   MEDIUM ANODIZED
TIN
16      Brand#13  STANDARD BURNISHED      50      3
        27      32      12      2       22      36
11      INDIA     0.0000000333
2       46      NICKEL   AMERICA
10      1993-10-01
18      312
1       82
13      express   accounts
7       FRANCE  MOZAMBIQUE
22      23      16      11      28      31      10
        13
3       HOUSEHOLD       1995-03-10
20      thistle   1995-01-01      EGYPT
```

qp1.5

```
21      EGYPT
15      1996-07-01
4       1997-01-01
6       1997-01-01      0.09      24
7       UNITED KINGDOM INDIA
16      Brand#53  MEDIUM POLISHED        3       42
        20      22      15      41      49      8
19      Brand#12  Brand#14  Brand#11  5       20      25
18      313
14      1995-12-01
22      29      20      13      24      33      22
        18
11      VIETNAM         0.0000000333
13      express   accounts
3       AUTOMOBILE      1995-03-26
1       90
2       34      COPPER   MIDDLE EAST
5       ASIA      1997-01-01
8       INDIA     ASIA     SMALL POLISHED TIN
20      goldenrod 1993-01-01      ROMANIA
12      FOB       TRUCK    1996-01-01
17      Brand#33  SM BAG
10      1994-07-01
9       dodger
```

qp1.6

```
10      1993-04-01
3       FURNITURE       1995-03-12
15      1994-04-01
13      express   accounts
6       1993-01-01      0.07      24
8       ALGERIA AFRICA   SMALL BURNISHED TIN
9       cornsilk
7       MOROCCO         ALGERIA
4       1994-10-01
11      INDONESIA       0.0000000333
22      17      22      29      18      15      12
        28
18      315
12      MAIL      FOB      1996-01-01
1       98
5       MIDDLE EAST     1993-01-01
16      Brand#33  ECONOMY BRUSHED        18      39
        8       16      12      23      15      45
2       21      STEEL    AMERICA
14      1996-03-01
19      Brand#14  Brand#52  Brand#15  10      10      21
20      rose      1997-01-01      INDONESIA
17      Brand#35  SM PACK
21      RUSSIA
```

qp1.7

```
18      312
8       PERU      AMERICA         STANDARD BRUSHED
TIN
20      cornflower 1995-01-01     UNITED STATES
21      KENYA
2       9       BRASS    MIDDLE EAST
4       1997-05-01
22      32      24      13      14      21      31
        20
17      Brand#32  SM DRUM
1       106
11      RUSSIA    0.0000000333
9       burnished
19      Brand#11  Brand#35  Brand#14  5       11      28
3       MACHINERY       1995-03-28
13      express   accounts
5       AFRICA    1993-01-01
7       GERMANY         PERU
10      1994-02-01
16      Brand#13  SMALL BURNISHED        17      27
        18      49      30      23      25      8
6       1993-01-01      0.04      25
14      1996-06-01
15      1996-11-01
12      RAIL      SHIP     1996-01-01
```

qp1.8

```
19      Brand#24  Brand#23  Brand#53  1       12      25
1       114
15      1994-08-01
17      Brand#33  LG BAG
5       AMERICA         1993-01-01
8       INDONESIA       ASIA     STANDARD PLATED
TIN
9       black
12      AIR       SHIP     1996-01-01
14      1996-09-01
7       UNITED STATES   INDONESIA
4       1995-02-01
3       FURNITURE       1995-03-14
```


© 2004 Hewlett Packard Company. All rights reserved

20	navajo	1994-01-01		JORDAN		
16	Brand#53	LARGE PLATED		43	7	9
	28	47	45	31	12	
6	1993-01-01		0.09	24		
22	13	27	17	23	15	14
	12					
10	1994-11-01					

13	express	accounts	
2	47	NICKEL	ASIA
21	FRANCE		
18	314		
11	IRAN	0.0000000333	

© 2004 Hewlett Packard Company. All rights reserved

Appendix F: Benchmark Scripts

2shut

```
#!/bin/ksh
. $FRAME_PATH/env

if [ "$1" = "abort" ]; then
for i in $SECONDARY_NODES
do
rsh $i -n $KIT_DIR/rshuta
done
sqlplus << !
connect / as sysdba
shutdown abort
exit
!
else
for i in $SECONDARY_NODES
do
rsh $i -n $KIT_DIR/rshut
done
sqlplus << !
connect / as sysdba
shutdown immediate
exit
!
fi
```

2start

```
#!/bin/ksh

. $FRAME_PATH/env
cud=`pwd`
cd $ORACLE_HOME/bin
cpall oracle
sleep 5
cd /home/oracle/tpch/dbs
cpall init_run.ora
cd $cud

if [[ "$1" = "" ]]; then
  nodes=$ALL_NODES
else
  nodes=$1
fi
for i in $nodes
do
echo Bringing up node $i
rsh $i -n $KIT_DIR/rstart &
sleep 5
done
wait
echo Nodes are up!
```

gen_seed.sh

```
#!/bin/ksh

SEED_FILE=$1

#Generate the seed
echo "Setting the random number seed"
PSEED=`date +%m:%d:%H:%M:%S | sed -e 's/://g'`
echo "Using ${PSEED} as seed0"
echo ${PSEED} >  $SEED_FILE
echo "Done setting the random number seed"
```

qexecpl.c

```
#ifdef RCSID
static char *RCSid =
   "$Header: qexecpl.c 17-oct-2001.09:29:47 mpoess Exp $ ";
#endif /* RCSID */

/* Copyright (c) 1999, 2001, Oracle Corporation.  All rights reserved.  */

/*

   NAME
     qexecpl.c - <one-line expansion of the name>

   DESCRIPTION
     SQL Execution Engine, Oracle v8,  OCI version

   PRIVATE FUNCTION(S)
     <list of static functions defined in .c file - with one-line descriptions>

   MODIFIED   (MM/DD/YY)
     mpoess    10/17/01 - add serialization level in SQLinit
     mpoess    02/22/01 - add linux changes
     mpoess    08/05/99 - make compile
     mpoess    11/13/98 - fix pddl statement
     pswong    02/19/97 - migrating to version 8
     pswong    04/02/96 - more polishing
     pswong    03/25/96 - polish up
     pswong    03/06/96 - created

*/

#include <stdio.h>
#include <string.h>
#include <setjmp.h>
#include <sys/param.h>
#include <errno.h>
#include <math.h>
#include <string.h>
#include <sys/types.h>
#include <time.h>
#include <stdlib.h>

#include "qexecpl.h"

/* Function Prototypes */

extern double gettime();

/* function prototypes from gen.c */

int get_statement();

/* Declare error handling functions */

void sql_error();

/* Other prototypes */
```


© 2004 Hewlett Packard Company. All rights reserved

```
int define_output_variables();
void process_select_list();
void usage();
void SQLinit();
void SQLexec();
void SQLexit();
void *memalloc();
void print_header();
void print_rows();
int OFEN();
void remove_newline();

char logname[UNAME_LEN];  /* username/passwd combo */
char *passwd;

double tr_start = 0.0;    /* query start time                  */
double tr_end = 0.0;      /* query end time                    */

double s_tr_start = 0.0;  /* statement start time              */
double s_tr_end = 0.0;    /* statement end time                */

/* For our purpose of timing, we will treat comments as delimiters */
/* for queries.   Thus, we will collect query timings whenever we  */
/* encounter a comment (of course not for the first comment in a   */
/* file).                                                          */

int end_flag = 0;         /* flag to indicate that we have reached */
                          /* the end of a query                    */

int stmt_cnt = 0;         /* Number of statements processed.     */
int qry_cnt = 0;          /* Number of query processed.          */

double product = 1.0;     /* cumulative product of query times   */
int rows_ret = 0;         /* the number of rows fetched          */
int num_sel_list = 0;     /* the number of select list item      */

long num_to_fetch = -1;   /* Number of rows to fetch. -1 means fetch all
*/

sltype slist[MAX_SEL_LIST];  /* Array for describing Select List
*/
dltype *dlist[MAX_SEL_LIST]; /* Array of ptrs for Defining Select
List   */

char stmt[SQL_LEN];       /* The SQL statement or comment line.  */
char qn[3];               /* Number of the query being executed  */
char qnp[3];              /* Number of the previous query executed  */
char cmnt[5000];          /* Buffer to save the comment.         */
#ifdef LINUX
FILE *qtemp;      /* fd for query template              */
FILE *logfile;   /* log and report files               */
FILE *rep;
#else
FILE *qtemp = stdin;      /* fd for query template              */
FILE *logfile = stdout;   /* log and report files               */
FILE *rep = stdout;
#endif
void *defbuf;             /* Buffer pointer for ODEFIN          */
int deflen = 0;           /* Size of data type for ODEFIN       */
int deftype = 1;          /* Oracle type number for ODEFIN      */

int pfmem = PFMEMSIZE;    /* Memory to prefetch rows            */

time_t tim;               /* To get wall clock time             */

/* OCI handles */

OCIEnv *tpcenv = NULL;
OCIServer *tpcsrv = NULL;
OCIError *errhp = NULL;
OCISvcCtx *tpcsvc = NULL;
OCISession *tpcusr = NULL;
OCIStmt *curq = NULL;
OCIStmt *cur_dml = NULL;
OCIStmt *cur_ddl = NULL;
OCIParam *tpcpar = NULL;

sword status = OCI_SUCCESS;  /* OCI return value */

/* usage: prints the usage of the program */

void usage() {

  fprintf(stderr,"\nUsage: qexec username/password [q<path name for
query template file>]\n");
  fprintf(stderr,"               [l<path name for log>] [r<path name for
reports>]\n\n");
  fprintf(stderr,"Options:\n");
  fprintf(stderr,"q<path for query>        : full path name for the query
template file.\n");
  fprintf(stderr,"                         (default is stdin)\n");
  fprintf(stderr,"l<path name for log>     : full path name for log
files\n");
  fprintf(stderr,"                         (default is stdout)\n");
  fprintf(stderr,"r<path name for reports>  : full path name for
reports\n");
  fprintf(stderr,"                         (default is stdout)\n");
  exit(-1);
}



/* type: 0 if environment handle is passed, 1 if error handle is passwd */

void sql_error(errhp,status,type)
    OCIError *errhp;
    sword status;
    sword type;
{
  char msg[2048];
  ub4 errcode;
  ub4 msglen;
  int i,j;

  switch(status) {
  case OCI_SUCCESS_WITH_INFO:
    fprintf(stderr, "Error: Statement returned with info.\n");
    if (type)
      (void) OCIErrorGet(errhp,1,NULL,(sb4*)&errcode,(text*)msg,
                                        2048,OCI_HTYPE_ERROR);
    else
      (void) OCIErrorGet(errhp,1,NULL,(sb4*)&errcode,(text*)msg,
                                        2048,OCI_HTYPE_ENV);
    fprintf(stderr,"%s\n",msg);
    break;
  case OCI_ERROR:
    fprintf(stderr, "Error: OCI call error.\n");
    if (type)
      (void) OCIErrorGet(errhp,1,NULL,(sb4*)&errcode,(text*)msg,
                                        2048,OCI_HTYPE_ERROR);
    else
      (void) OCIErrorGet(errhp,1,NULL,(sb4*)&errcode,(text*)msg,
                                        2048,OCI_HTYPE_ENV);
    fprintf(stderr,"%s\n",msg);
    break;
  case OCI_INVALID_HANDLE:
    fprintf(stderr, "Error: Invalid Handle.\n");
    if (type)
      (void) OCIErrorGet(errhp,1,NULL,(sb4*)&errcode,(text*)msg,
                                        2048,OCI_HTYPE_ERROR);
```


© 2004 Hewlett Packard Company. All rights reserved

```
    else
      (void) OCIErrorGet(errhp,1,NULL,(sb4*)&errcode,(text*)msg,
                                        2048,OCI_HTYPE_ENV);
    fprintf(stderr,"%s\n",msg);
    break;
  }

  /* Rollback just in case */

  (void) OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);

  fprintf(stderr, "Exiting Oracle...\n");
  fflush(stderr);

  SQLexit();

  exit(1);
}

#ifdef LINUX
int main(argc,argv)
#else
void main(argc,argv)
#endif
     int argc;
     char *argv[];
{

  int i,pos,pos2;
  int retcode;      /* Return code for get_statement */
#ifdef LINUX
  logfile=fopen("/dev/stdout","w");
  qtemp=fopen("/dev/stdin","rw");
  rep=fopen("/dev/stdout","w");
#endif
  /* Initialize some variables */

  if ((argc > 5) || (argc < 2)) {
    usage();
  }

  /* argv[1] -- User and Password for Database */

  strcpy(logname, argv[1]);

  /* Process  optional parameters */

  argc -= 1;
  argv += 1;

  while(--argc) {
    ++argv;
    switch(argv[0][0]) {
    case 'q':
      if ((qtemp = fopen(++(argv[0]),"r")) == NULL) {
                fprintf(stderr,"Unable to open file '%s'\n", argv[0]);
                fprintf(stderr,"%s: %s\n", argv[0], strerror(errno));
                exit(-1);
      }
      break;
    case 'r':
      if ((rep = fopen(++(argv[0]),"a")) == NULL) {
                fprintf(stderr,"Unable to open file '%s'\n", argv[0]);
                fprintf(stderr,"%s: %s\n", argv[0], strerror(errno));
                exit(-1);
      }
      break;
    case 'l':
      if ((logfile = fopen(++(argv[0]),"a")) == NULL) {
                fprintf(stderr,"Unable to open file '%s'\n", argv[0]);
                fprintf(stderr,"%s: %s\n", argv[0], strerror(errno));
                exit(-1);
      }
      break;
    default:
      fprintf(stderr,"Invalid Option: %c\n", argv[0][0]);
      usage();
      break;
    }
  }

  /* Do some initialization and establish connection with the database */

  SQLinit();

  /* May want to add some triggering mechanism here */

  time(&tim);
  fprintf(logfile, "Begin Execution at %s\n\n", ctime(&tim));
  fprintf(rep, "Begin Executing this Stream at %s\n\n", ctime(&tim));
  /* Get the next statement and start processing it */

  while ((retcode = get_statement()) > 0) {

    switch (retcode) {

      /* If this is a comment, skips it */
    case COMMENT:
      /*if (end_flag) {
                end_flag = 0;  */    /* reset query end flag */
                /* save the comment so that we can print it out later on  */
                /* strcpy(cmnt, stmt);
                break;
      } */
      if (stmt[3]== '@') {
        pos=4;
        strcpy(qnp,qn);
        while (stmt[pos] != ')') {
          pos++;
        }
        pos2=0;
        pos++;
        while (stmt[pos] != '.') {
          /*printf ("qn %d %c \n",pos2,stmt[pos]);*/
          qn[pos2]=stmt[pos];
          pos2++;
          pos++;
        }
        qn[pos2] = 0;
        /* printf("found a new query: %s\n",qn); */
      }
      /* save the comment so that we can print it out later on  */
      strcat(cmnt, stmt);
      break;

      /* if this is a set_row_fetch command */
    case SET_FETCHROW:
      fprintf(logfile,"Setting the number of rows to fetch to: %ld\n\n",
                    num_to_fetch);
      break;

      /* if this is a SQL statement */
    case SQL_STMT:

      /* Executes the query */
      SQLexec();

      stmt_cnt++;
      qry_cnt++;
      fflush(rep);
```


© 2004 Hewlett Packard Company. All rights reserved

```
    fflush(logfile);
    /*
    fprintf(logfile,"\nStatement Started at %.2f\n", s_tr_start);
    fprintf(logfile,"Statement Ended at %.2f\n", s_tr_end);

    fprintf(logfile,"Statement Processed in %.2f seconds.\n",
                (s_tr_end - s_tr_start));
    fprintf(rep, "Query %s: Execution Time: %.2f started %.2f ended
%.2f\n",
              qn,(s_tr_end - s_tr_start)s_tr_start,s_tr_end);
    fflush(rep);
    fflush(logfile);*/
    break;

    /* Should never reach here */
   default:
    fprintf(stderr, "Invalid statement type!!\n");
    SQLexit();
    break;
   }
  }

  /* Get Timing for the last query */

  tr_end = gettime();

  fprintf(logfile,"Query Processed in %.2f seconds.\n\n",(tr_end -
s_tr_start));

  /* print comments for this query that we have saved */

  /* fprintf(logfile, "%s\n", cmnt); */

  /* fprintf(rep, "Query %s : Execution time %.2f\n", qn,(tr_end -
s_tr_start));*/
  fprintf(rep, "Query %s: Execution Time: %.2f started %.2f ended
%.2f\n",
              qn,(tr_end - s_tr_start),s_tr_start,tr_end);

  time(&tim);
  fprintf(logfile,"\nEnded Executing this Stream at %s\n", ctime(&tim));
  fprintf(logfile,"\nStream Started at %.2f\n", tr_start);
  fprintf(logfile,"Stream Ended at %.2f\n", tr_end);
  fprintf(logfile,"Stream Processed in %.2f seconds\n\n",(tr_end -
tr_start));

  fprintf(rep,"\nEnded Executing this Stream at %s\n", ctime(&tim));
  fprintf(rep,"\nStream Started at %.2f\n", tr_start);
  fprintf(rep,"Stream Ended at %.2f\n", tr_end);
  fprintf(rep,"Stream Processed in %.2f seconds\n\n",
              (tr_end - tr_start));

  fprintf(logfile, "\nSQL statements processed: %d\n", stmt_cnt);
  /*fprintf(logfile, "Queries processed: %d\n", qry_cnt);*/

  fflush(rep);
  fflush(logfile);

  /* Close the query template file */

  fclose(qtemp);

  /*  Disconnect from ORACLE. */

  SQLexit();
  exit(0);
}


/* SQLinit(): Perform initialization tasks.                    */
/*          Logs on to Oracle, opens some files and open a cursor for */
/*          later use.                                          */

void SQLinit() {

  int i;

  /* preallocate MAX_PREALLOC members of the dlist array
*/
  /* initializes others to NULL so that we can determine who to free later
*/

  for (i=0; i<MAX_SEL_LIST; i++) {
    if (i < MAX_PREALLOC) {
      dlist[i] = (dltype *) memalloc (sizeof(dltype));
      dlist[i]->defhdl = NULL;
/*      OCIhalloc(curq,&(dlist[i]->defhdl),OCI_HTYPE_DEFINE); */
    }
    else
      dlist[i] = NULL;
  }

  /* Connect to ORACLE.  Program will call sql_error()        */
  /* if an error occurs in connecting to the default database. */

  (void) OCIInitialize(OCI_DEFAULT,(dvoid *)0,0,0,0);

  if((status=OCIEnvInit((OCIEnv **)&tpcenv,OCI_DEFAULT,0,(dvoid
**)0)) !=
     OCI_SUCCESS)
    sql_error(tpcenv, status, 0);

  OCIhalloc(tpcenv,&errhp,OCI_HTYPE_ERROR);
  OCIhalloc(tpcenv,&curq,OCI_HTYPE_STMT);
  OCIhalloc(tpcenv,&cur_dml,OCI_HTYPE_STMT);
  OCIhalloc(tpcenv,&cur_ddl,OCI_HTYPE_STMT);
  OCIhalloc(tpcenv,&tpcsvc,OCI_HTYPE_SVCCTX);
  OCIhalloc(tpcenv,&tpcsrv,OCI_HTYPE_SERVER);
  OCIhalloc(tpcenv,&tpcusr,OCI_HTYPE_SESSION);

  /* get username and password */

  passwd = strchr(logname, '/');
  *passwd = '\0';
  passwd++;

  if ((status = OCIServerAttach(tpcsrv,errhp,(text
*)0,0,OCI_DEFAULT)) != OCI_SUCCESS)
    sql_error(errhp,status,1);


OCIaset(tpcsvc,OCI_HTYPE_SVCCTX,tpcsrv,0,OCI_ATTR_SERVER
,errhp);

OCIaset(tpcusr,OCI_HTYPE_SESSION,logname,strlen(logname),OCI_
ATTR_USERNAME,
                errhp);

OCIaset(tpcusr,OCI_HTYPE_SESSION,passwd,strlen(passwd),OCI_A
TTR_PASSWORD,
                errhp);

  if ((status = OCISessionBegin(tpcsvc, errhp, tpcusr,
OCI_CRED_RDBMS,
                                        OCI_DEFAULT)) !=
OCI_SUCCESS)
    sql_error(errhp,status,1);
```


© 2004 Hewlett Packard Company. All rights reserved

```
OCIaset(tpcsvc,OCI_HTYPE_SVCCTX,tpcusr,0,OCI_ATTR_SESSIO
N,errhp);

/*
  if ((status=OCILogon((OCIEnv *)tpcenv,(OCIError
*)errhp,(OCISvcCtx *)tpcsvc,
                        (text *)logname, strlen(logname), (text
*)passwd,
                        strlen(passwd), (text *) 0, 0)) !=
OCI_SUCCESS)
    sql_error(errhp, status, 1);
*/
  printf("\nConnected to ORACLE as user: %s\n\n", logname);

}



/* SQLexec()  Executes the SQL statement.                    */
/*          Parse the SQL statement.                    */
/*          If DDL or DML statements, execute right away.          */
/*          Else describe and define select list outputs,          */
/*              execute and fetch results.                    */

void SQLexec()
{
  int i;
  ub2 stmttyp = OCI_STMT_SELECT;     /* default is a SELECT
statement */

  /* Clause 5.3.6.2: QI(i,s) is the time between the first character */
  /*              of this query text is submitted and the first   */
  /*              character of the next query text is submitted.  */

  if (qry_cnt) {
    time(&tim);
    s_tr_end = gettime();
    fprintf(logfile,"Query Processed in %.2f seconds.\n\n",
     (s_tr_end - s_tr_start));

    /* print comments for this query that we have saved */

    /* fprintf(logfile, "%s\n", cmnt); */

    /*fprintf(rep, "Query %s : Execution time %.2f\n", qnp,(s_tr_end -
s_tr_start));*/
    fprintf(rep, "Query %s: Execution Time: %.2f started %.2f ended
%.2f\n",
                qnp,(s_tr_end - s_tr_start),s_tr_start,s_tr_end);

    /* Let's fflush stuff so that we can see what's going on */

    /* Fix for Q15 */
    strcpy(qnp,qn);

    fflush(logfile);
    fflush(rep);
  }
  else
    tr_start = gettime();

  s_tr_start = gettime();

  /* prepare the statement */

  if ((status = OCIStmtPrepare(curq, errhp, (text*) stmt, (ub4)
strlen(stmt),
                                    OCI_NTV_SYNTAX,
OCI_DEFAULT)) != OCI_SUCCESS)
    sql_error(errhp,status,1);

  /* Prints the query text and comment to the logfile */

  fprintf(logfile, "\n%s\n", cmnt);
  cmnt[0]=0;
  fprintf(logfile, "\n%s\n", stmt);

  /* if this is a DDL or DML statement, execute it right away */
  /* only worries about SELECT statements right now, cannot   */
  /* execute a stored PL/SQL procedure in thie version        */


OCIaget(curq,OCI_HTYPE_STMT,&stmttyp,NULL,OCI_ATTR_STM
T_TYPE,errhp);

  if (stmttyp != OCI_STMT_SELECT) {
    OCIsexec(tpcsvc,curq,errhp,1);
    return;
  }

  /* otherwise, this is a select statement */
  /* Describe and define output variables  */

  /* first let's execute it to get the select-list definition */

  OCIaset(curq, OCI_HTYPE_STMT, &pfmem, 0,
OCI_ATTR_PREFETCH_MEMORY, errhp);

  OCIsexec(tpcsvc,curq,errhp,0);

  num_sel_list = define_output_variables();

  /* Executes the query and fetches the rows */

  (void) process_select_list(num_sel_list);

  /* Need to get the number of rows fetched first   */
  /* since the following statments will screw it up */


OCIaget(curq,OCI_HTYPE_STMT,&rows_ret,NULL,OCI_ATTR_RO
W_COUNT,errhp);

  /* To control memory usage, let's free up the extra dlist entries */
  /* that we have allocated.                                        */

  i=MAX_PREALLOC;
  while(dlist[i] != NULL) {
    free(dlist[i]);
    dlist[i++] = NULL;
  }

  /* reset set_fetchrows */

  num_to_fetch = -1;

}


void SQLexit() {

  int i;

  OCILogoff(tpcsvc,errhp);
  OCIhfree(tpcenv,OCI_HTYPE_STMT);
  OCIhfree(tpcsvc,OCI_HTYPE_SVCCTX);
  OCIhfree(tpcsrv,OCI_HTYPE_SERVER);
```


© 2004 Hewlett Packard Company. All rights reserved

```
  OCIhfree(tpcusr,OCI_HTYPE_SESSION);

  /* free all memory */

  for (i=0; i<MAX_SEL_LIST; i++) {
    if (dlist[i] != NULL) {
      free(dlist[i]);
    }
  }

  /* Flush all output */

  fflush(rep);
  fflush(logfile);

}


/* define_output_variables(): Describe and define select-list items for */
/*                            a query statement.                         */
/*                            Returns the number of select-list items    */
/*                            for this query.                            */

int define_output_variables()
{

  int i;
  int retflag = 0;

  for (i=0; i<MAX_SEL_LIST; i++) {

    slist[i].buflen = MAX_COLNAME_SIZE;

    if (OCIParamGet(curq, OCI_HTYPE_STMT, errhp, (dvoid **)
&tpcpar,
                                        POS(i)) != OCI_SUCCESS)
      break;


    /* dsize and nullok fields of dlist not used */

    OCIaget(tpcpar, OCI_DTYPE_PARAM, &(slist[i].dbsize),
                NULL, OCI_ATTR_DATA_SIZE, errhp);
    OCIaget(tpcpar, OCI_DTYPE_PARAM, &(slist[i].dbtype),
                NULL, OCI_ATTR_DATA_TYPE, errhp);
    OCIaget(tpcpar, OCI_DTYPE_PARAM, &(slist[i].buf),
                &(slist[i].buflen), OCI_ATTR_NAME, errhp);
    OCIaget(tpcpar, OCI_DTYPE_PARAM, &(slist[i].precision),
                NULL, OCI_ATTR_PRECISION, errhp);
    OCIaget(tpcpar, OCI_DTYPE_PARAM, &(slist[i].scale),
                NULL, OCI_ATTR_SCALE, errhp);

    /* For formatting purpose, remove trailing blanks in select-list name.
*/

/*
    if (slist[i].buflen < MAX_COLNAME_SIZE)
      (slist[i].buf)[slist[i].buflen] = '\0';
*/
    /* Well, we need to allocate for entries for dlist */

    if (i >= MAX_PREALLOC) {
      dlist[i] = (dltype *) memalloc(sizeof(dltype));
      dlist[i]->defhdl = NULL;
    }

    /* Let's check the sizes and types for this select list item */

    switch (slist[i].dbtype) {

      case OCI_TYPECODE_NUMBER:

        /* The odescr will not give a good estimate to the scale if */
        /* no scale was given in the Oracle table definition.       */

#ifdef HAVE_SCALE
        if (slist[i].scale != 0) {
                defbuf = (double *) dlist[i]->fbuf;
                deflen = FLT;
                deftype = OCI_TYPECODE_DOUBLE;
                slist[i].dbtype = OCI_TYPECODE_DOUBLE;
        } else {
                defbuf = (int *) dlist[i]->ibuf;
                deflen = INT;
                deftype = OCI_TYPECODE_INTEGER;
                slist[i].dbtype = OCI_TYPECODE_INTEGER;
        }
#else
        defbuf = (double *) dlist[i]->fbuf;
        deflen = FLT;
        deftype = OCI_TYPECODE_FLOAT;
        slist[i].dbtype = OCI_TYPECODE_FLOAT;
#endif /* HAVE_SCALE */

        break;

      default:

        /* default is character string */

        defbuf = (char **) dlist[i]->sbuf;
        deflen = MAX_STR_LEN;
        deftype = SQLT_STR;
/*      deftype = OCI_TYPECODE_CHAR; */
        break;
      }

      /* Define the column */

      if ((status=OCIDefineByPos(curq,&(dlist[i]->defhdl),errhp,POS(i),
                                        defbuf,deflen,deftype,NULL,
                                        dlist[i]-
>rlen,NULL,OCI_DEFAULT))!=OCI_SUCCESS)
        sql_error(errhp,status,1);
    }
    return i;
  }


/* process_select_list(): Fetch rows from a query.                    */

void process_select_list(num)
    int num;       /* number of select list items */
{

  int i,j;
  int ntf;
  int num_so_far;
  sword stats = OCI_SUCCESS;

  /* Print the headers for the query execution result */

  print_header(num);

  /* See if we need to limit the rows to fetch */

  ntf = (num_to_fetch >= 0) ? num_to_fetch : MAX_ARRAY;

  /* Fetch the rows and print them out */
```


© 2004 Hewlett Packard Company. All rights reserved

```
  if ((ntf > MAX_ARRAY) || (num_to_fetch == -1)) {

   stats = OCIStmtFetch(curq, errhp, MAX_ARRAY,
OCI_FETCH_NEXT, OCI_DEFAULT);


OCIaget(curq,OCI_HTYPE_STMT,&rows_ret,NULL,OCI_ATTR_RO
W_COUNT,errhp);

   print_rows(num,rows_ret);

   /* To avoid 1022 from OFEN */
   /* More rows to fetch...   */

   if (stats != OCI_NO_DATA) {
    if (num_to_fetch == -1) {
            while ((stats =
OCIStmtFetch(curq,errhp,MAX_ARRAY,OCI_FETCH_NEXT,
                                                OCI_DEFAULT)) ==
OCI_SUCCESS) {
              OCIaget(curq,OCI_HTYPE_STMT,&num_so_far,NULL,
                          OCI_ATTR_ROW_COUNT,errhp);
              print_rows(num,(num_so_far-rows_ret));
              rows_ret = num_so_far;
            }
            /* Print the final rows */
            OCIaget(curq,OCI_HTYPE_STMT,&num_so_far,NULL,
                        OCI_ATTR_ROW_COUNT,errhp);
            print_rows(num,(num_so_far-rows_ret));
            rows_ret = num_so_far;
    } else {
            ntf -= MAX_ARRAY;

            while ((stats = OCIStmtFetch(curq,errhp,
                                                ((ntf>MAX_ARRAY) ?
MAX_ARRAY:ntf),
                                                OCI_FETCH_NEXT,
OCI_DEFAULT)) ==
                 OCI_SUCCESS) {
              ntf -= MAX_ARRAY;
              OCIaget(curq,OCI_HTYPE_STMT,&num_so_far,NULL,
                          OCI_ATTR_ROW_COUNT,errhp);
              print_rows(num,(num_so_far-rows_ret));
              rows_ret = num_so_far;
              if (ntf <= 0) break;
            }
            OCIaget(curq,OCI_HTYPE_STMT,&num_so_far,NULL,
                        OCI_ATTR_ROW_COUNT,errhp);
            print_rows(num,(num_so_far-rows_ret));
            rows_ret = num_so_far;
    }
   }
  } else {
   OCIStmtFetch(curq, errhp, ntf, OCI_FETCH_NEXT,
OCI_DEFAULT);

OCIaget(curq,OCI_HTYPE_STMT,&rows_ret,NULL,OCI_ATTR_RO
W_COUNT,errhp);
   print_rows(num,rows_ret);
  }

  fprintf(logfile,"\n\n%d %s processed.\n", rows_ret,
              rows_ret == 1 ? "row" : "rows");

}


int get_statement()
{
  char line[128];
  char *pos, *str;

  /* Reset statement buffer */

  stmt[0] = '\0';

  while (fgets(line, 127, qtemp) != NULL) {

   /* skip blank lines */
   if (line[0] == '\n')
    continue;

   /* remove blanks */

   str = line;

   while (*str == ' ') str++;

   /* Let's get the line together first */

   strcat(stmt, str);

   /* if this is a comment line */
   if ((str[0] == '-') && (str[1] == '-'))
    return COMMENT;

   /* see if this is a set_fetchrows line */
   if (strncmp(str, "set_fetchrows", 13) == 0) {
    pos = strchr(str, ';');
    *pos = '\0';
    pos = strchr(str, '=');
    num_to_fetch = atol(++pos);
    return SET_FETCHROW;
   }

   /* if this is the end of the current statement */
   if ((pos = strchr(stmt, ';')) != NULL) {
    *pos = '\0';
    return SQL_STMT;
   }
  }
  return END_OF_FILE;
}


/* memalloc(): Allocates memory, exit program if we have a problem. */

void *memalloc(size)
    int size;
{

  void *tmp;

  if ((tmp = (void *) malloc(size)) == NULL) {
   fprintf(stderr, "Error in malloc\n");
   SQLexit();
   return NULL;      /* should never reach here */
  } else {
   return tmp;
  }
}



void print_header(nsel)
    int nsel;         /* Number of select list items */
{

  int i, diff;
```


© 2004 Hewlett Packard Company. All rights reserved

```
  char colname[MAX_COLNAME_SIZE];
  int len = 0;          /* Running column length */
  int cwid = 0;

  fprintf(logfile, "\n");

  for (i=0; i<nsel; i++) {

   /* extract the column name */

   strncpy((char *)colname, (char *)slist[i].buf, slist[i].buflen);
   colname[slist[i].buflen] = '\0';

   /* format the output a little */

   cwid = MAX(slist[i].dbsize, slist[i].buflen);

   /* do a little bit of formatting */

   if (cwid > 80) {
     fprintf(logfile,"\n");
     len = 0;
   } else if ((len += cwid) > 80) {
     fprintf(logfile,"\n");
     len = cwid;
   }
#ifdef FORMAT1
   if ((slist[i].dbtype == INT_TYPE) || (slist[i].dbtype == FLT_TYPE))
     fprintf(logfile, "%*s ", cwid, slist[i].buf);
   else  /* string type */
     fprintf(logfile, "%*s ", -cwid, slist[i].buf);
#else
     fprintf(logfile, "%*s ", -cwid, colname);
#endif /* FORMAT1 */
  }

  fprintf(logfile,"\n");
}


void print_rows(ncol, nrow)
    int ncol;
    int nrow;
{

  int i,j;
  int len;
  int diff;
  int cwid;

  for (i=0;i<nrow;i++) {

   len = 0;

   for (j=0;j<ncol;j++) {

    cwid = MAX(slist[j].dbsize, slist[j].buflen);

    /* do a little bit of formatting */

    if (cwid > 80) {
            fprintf(logfile,"\n");
            len = 0;
    } else if ((len += cwid) > 80) {
            fprintf(logfile,"\n");
            len = cwid;
    }

    switch(slist[j].dbtype) {
    case INT_TYPE:
#ifdef HAVE_SCALE
            fprintf(logfile, "%*ld|", cwid,      (dlist[j]->ibuf)[i]);
            break;
#endif /* HAVE_SCALE */
     case FLT_TYPE:
#ifdef FORMAT1
            fprintf(logfile,"%*.2f ", cwid, (dlist[j]->fbuf)[i]);
#else
            fprintf(logfile,"%*.2f ", -cwid, (dlist[j]->fbuf)[i]);
#endif /* FORMAT1 */
            break;
     default:
            fprintf(logfile, "%*s ", -(cwid), (dlist[j]->sbuf)[i]);
            break;
     }
    }
    fprintf(logfile, "\n");
  }
}


/* remove_newline(): Remove newline character from str. */

void remove_newline(str)
    char *str;
{

  char *p;

  while ((p = strchr(str,'\n')) != NULL)
    *p = ' ';
}
```

qexecpl.h

```
/*
 * $Header: qexecpl.h 13-nov-2001.17:52:35 mpoess Exp $
 */

/* Copyright (c) 1999, 2001, Oracle Corporation.  All rights reserved.  */

/* NOTE:  See 'header_template.doc' in the 'doc' dve under the 'forms'
     directory for the header file template that includes instructions.
*/

/*
   NAME
     qexecpl.h

   DESCRIPTION
     SQL statement execution front-end header file.

   PUBLIC FUNCTION(S)
     <list of external functions declared/defined - with one-line
descriptions>

   PRIVATE FUNCTION(S)
     <list of static functions defined in .c file - with one-line descriptions>

   EXAMPLES

   NOTES
     <other useful comments, qualifications, etc.>

   MODIFIED   (MM/DD/YY)
   mpoess     11/13/01 - change DOP to 84 for DML and DDL
   mpoess     02/22/01 - add linux changes
   mpoess     08/05/99 - make compile
   mpoess     07/15/99 - Creation
   mpoess     07/15/99 - Creation
```


© 2004 Hewlett Packard Company. All rights reserved

```
*/

/*
# ifndef S_ORACLE
#  include <s.h>
# endif
*/
#ifndef QSTREAMPL_H

#define QSTREAMPL_H

#include <stdio.h>
#include <string.h>
#include <sys/param.h>
#include <sys/types.h>
#include <time.h>
#include <errno.h>
#include <math.h>

#include <oratypes.h>

#include <oratypes.h>

#ifndef OCIDFN
#include <ocidfn.h>
#endif /* OCIDFN */

#ifndef OCI_ORACLE
#include <oci.h>
#endif /* OCI_ORACLE */
/*
#ifdef __STDC__
#include <ociapr.h>
#else
#include <ocikpr.h>
#endif *//* __STDC__ */

/* some basic definitions */

#define UNAME_LEN 64
#define MAX_FILE_PATH_LEN 128

#ifndef TRUE
#define TRUE 1
#endif /* TRUE */

#ifndef FALSE
#define FALSE 1
#endif /* FALSE */
#ifndef LINUX
#define MAX(x,y) ((x >= y) ? x : y)
#define MIN(x,y) ((x <= y) ? x : y)
#endif
/* defines and typedefs for parsing */

#define CRT_TBL  1
#define INS_STMT 3
#define SEL_STMT 4
#define UPD_STMT 5
#define DRP_VIEW 7
#define DRP_TBL  8
#define DEL_STMT 9
#define CRT_VIEW 10

/* defines and typedefs for query description */

#define MAX_COLNAME_SIZE 32   /* Maximum length of Column
name  */
#define MAX_SEL_LIST 16       /* Maximum items on a select list */
#define END_OF_LIST 1007      /* Error code when we reach the end of
the */
                              /* select list.                       */

/* types for describe */

#define CHAR_TYPE 1
#define NUM_TYPE  2
#define INT_TYPE  3
#define FLT_TYPE  4
#define STR_TYPE  5
#define DATE_TYPE 12

#define NUMWIDTH 16           /* Width of the numeric fields    */

#define POS(i) (i+1)          /* The position is 1...n instead  */
#define IND(i) (i-1)          /* of 0..n-1 as in an array.      */

typedef struct des
{
  ub2 dbsize;
  ub4 buflen;
/*  sb2 dsize;  */
  sb4 scale;
/*  sb2 nullok; */
  OCITypeCode dbtype;
/*  text buf[MAX_COLNAME_SIZE]; */
  text *buf;
  ub1 precision;
} sltype;


/* defines and typedefs for query select list definition */

#define MAX_ARRAY 50       /* Maximum array size for array fetch */
#define PFMEMSIZE 65536    /* Memory size of prefetch buffer     */

#define MAX_STR_LEN 256    /* Maximum size for string variables
*/
#define MAX_PREALLOC 8     /* Maximum number of preallocated
select list */
                           /* definitions.                       */

#define INT sizeof(long)
#define STR sizeof(char)
#define FLT sizeof(double)

#define FLTP (double *)
#define INTP (long *)
#define STRP (char **)

typedef struct def
{
  long ibuf[MAX_ARRAY];
  double fbuf[MAX_ARRAY];
  char sbuf[MAX_ARRAY][MAX_STR_LEN];
  ub2 rlen[MAX_ARRAY];        /* return length     */
  OCIDefine *defhdl;
} dltype;

extern int errno;

#define SQL_LEN 2048

#ifndef NULL
#define NULL 0
#endif

#ifndef NULLP
```


© 2004 Hewlett Packard Company. All rights reserved

```
# define NULLP (void *)NULL
#endif /* NULLP */

#ifndef DISCARD
# define DISCARD (void)
#endif

#ifndef sword
# define sword int
#endif

#ifndef ub1
#define ub1 unsigned char
#endif

#define NA              -1      /* ANSI SQL NULL */
#define VER7            2
#define NOT_SERIALIZABLE  8177  /* ORA-08177: transaction not
serializable */

#define ADR(object) ((ub1 *)&(object))
#define SIZ(object) ((sword)sizeof(object))
#define SID(sid) ((sid == -1) ? 0 : sid)

/* For get_statement */

#define END_OF_FILE -1
#define COMMENT 1
#define SQL_STMT   2
#define SET_FETCHROW 3


#define OCIhalloc(envh,hndl,htyp) \
   if((status=OCIHandleAlloc((dvoid *)envh,(dvoid
**)hndl,htyp,0,(dvoid **)0))!=OCI_SUCCESS) \
      sql_error(envh,status,0); \
   else \
      DISCARD 0

#define OCIhfree(hndl,htyp) \
   if((status=OCIHandleFree((dvoid *)hndl,htyp)) == OCI_SUCCESS) \
      fprintf(stderr, "Error freeing handle of type %d\n", htyp)

#define OCIaget(hndl,htyp,attp,size,atyp,errh) \
   if((status=OCIAttrGet((dvoid *)hndl,htyp,(dvoid *)attp,(dvoid
*)size,atyp,errh)) != OCI_SUCCESS) \
      sql_error(errh,status,1); \
   else \
      DISCARD 0

#define OCIaset(hndl,htyp,attp,size,atyp,errh) \
   if((status=OCIAttrSet((dvoid *)hndl,htyp,(dvoid
*)attp,size,atyp,errh)) != OCI_SUCCESS) \
      sql_error(errh,status,1); \
   else \
      DISCARD 0

#define OCIsexec(svch,stmh,errh,iter) \

if((status=OCIStmtExecute(svch,stmh,errh,iter,0,NULL,NULL,OCI_DE
FAULT)) != OCI_SUCCESS) \
      sql_error(errh,status,1); \
   else \
      DISCARD 0


#define ISOTXT "alter session set isolation_level = serializable"
#define PDMLTXT "alter session force parallel dml parallel (degree
84)"
#define PDDLTXT "alter session force parallel ddl parallel (degree 84)"

#endif /* QSTREAMPL_H */
```

runTPCH

```
#!/bin/ksh
. $KIT_DIR/env

ECHO=echo

sqlplus=$ORACLE_HOME/bin/sqlplus
GTIME=${KIT_DIR}/utils/gtime

RUN_ID_FILE=${KIT_DIR}/audit/r_id

if [ ! -f $RUN_ID_FILE ]
then
  echo "0" > $RUN_ID_FILE
fi

RUN_ID=`cat $RUN_ID_FILE`
RUN_ID=`expr $RUN_ID + 1`
echo $RUN_ID > $RUN_ID_FILE

OUT_DIR=${KIT_DIR}/audit/tests/${RUN_ID}
RDB_TABLES=${OUT_DIR}/rdbtablest
FIRST_TEN=${OUT_DIR}/firstten

if [ ! -d $OUT_DIR ]
then
  mkdir $OUT_DIR
fi

SCRIPT_LOG_FILE=${OUT_DIR}/main.out

2start >> $SCRIPT_LOG_FILE
ckpnt.sh

echo "Start: dbtables.sql and count.sql" >> $SCRIPT_LOG_FILE
$sqlplus ${DATABASE_USER} @$KIT_DIR/audit/dbtables >
${RDB_TABLES} 2>&1
$sqlplus ${DATABASE_USER} @$KIT_DIR/audit/firstten >
${FIRST_TEN} 2>&1
echo "End: dbtables.sql and count.sql `date`" >> $SCRIPT_LOG_FILE


runTPCHpt ${SCALE_FACTOR} 1 ${RUN_ID}
sleep 10
ckpnt.sh
runTPCHpt ${SCALE_FACTOR} 2 ${RUN_ID}

sleep 600
2shut >> $SCRIPT_LOG_FILE

cp $ORACLE_HOME/rdbms/log/alert_${ORACLE_SID}.log
$OUT_DIR

echo "End TPC-H Benchmark SEQUENCE NUMBER: $RUN_ID
`date`" >> $SCRIPT_LOG_FILE
```

runTPCH.load

```
#!/bin/ksh
. $KIT_DIR/env

ECHO=echo
```


© 2004 Hewlett Packard Company. All rights reserved

```
sqlplus=$ORACLE_HOME/bin/sqlplus
GTIME=${KIT_DIR}/utils/gtime

RUN_ID_FILE=${KIT_DIR}/audit/r_id

if [ ! -f $RUN_ID_FILE ]
then
  echo "0" > $RUN_ID_FILE
fi

RUN_ID=`cat $RUN_ID_FILE`
RUN_ID=`expr $RUN_ID + 1`
echo $RUN_ID > $RUN_ID_FILE

OUT_DIR=${KIT_DIR}/audit/tests/${RUN_ID}
if [ ! -d $OUT_DIR ]
then
  mkdir $OUT_DIR
fi

SCRIPT_LOG_FILE=${OUT_DIR}/main.out
RDB_TABLES=${OUT_DIR}/rdbtablest
FIRST_TEN=${OUT_DIR}/firstten
LD1DBCRE=${OUT_DIR}/Ld1dbcre
LD2SCTSO=${OUT_DIR}/Ld2sctso
LD3DAPOP=${OUT_DIR}/Ld3dapop
LD4IXCRE=${OUT_DIR}/Ld4ixcre
LD5ANLYZ=${OUT_DIR}/Ld5anlyz

echo Start TPC-H Benchmark SEQUENCE NUMBER: $RUN_ID >
$SCRIPT_LOG_FILE
echo >> $SCRIPT_LOG_FILE
echo "Starting a new Oracle log file:
$ORACLE_HOME/rdbms/log/alert_${ORACLE_SID}.log" >>
$SCRIPT_LOG_FILE
echo >> $SCRIPT_LOG_FILE

mv $ORACLE_HOME/rdbms/log/alert_${ORACLE_SID}.log
$ORACLE_HOME/rdbms/log/alert_${ORACLE_SID}.log.preAudit.$R
UN_ID
touch $ORACLE_HOME/rdbms/log/alert_${ORACLE_SID}.log

stime=`$GTIME`
2start
echo "Start: timed load portion `date`" >> $SCRIPT_LOG_FILE
dapop.sh > $LD3DAPOP
ixcre.sh  > $LD4IXCRE
anlyz.sh > $LD5ANLYZ
2shut
2start
ckpnt.sh
2shut

echo "End: timed load portion `date`" >> $SCRIPT_LOG_FILE
$KIT_DIR/audit/gen_seed.sh $KIT_DIR/audit/seed
echo Generated seed: `cat $KIT_DIR/audit/seed` >>
$SCRIPT_LOG_FILE

exit
```

runTPCHpt

```
#!/bin/ksh
. $KIT_DIR/env
#set -x
#ECHO=/bin/echo
SCRIPT_DIR=${KIT_DIR}/scripts
SQL_DIR=${KIT_DIR}/sql
UPD_DIR=${KIT_DIR}/update
SRC_DIR=${KIT_DIR}/utils
QRY_DIR=${KIT_DIR}/queries # this is the location of the query
template file
QGEN_DIR=${KIT_DIR}/dbgen
QGEN=${QGEN_DIR}/qgen
QEXEC=${SRC_DIR}

DSS_QUERY=${KIT_DIR}/queries
export DSS_QUERY

UPD_SQL=${UPD_DIR}/sql
UPD_SPT=${UPD_DIR}/scripts
UPD_SRC=${UPD_DIR}/source
UPD_DAT=${UPD_DIR}/data

TPCD_BIN=${KIT_DIR}/audit/bin

GTIME=${SRC_DIR}/gtime
SEED_FILE=${KIT_DIR}/audit/seed

DF=/dev/null
HID=1
INTERVAL=60
COUNT=1200

# The defaults

QPROG=${QEXEC}/qexec

usage () {

echo " "
echo "Usage: $0 [-p <program for query stream>] [-u1 <program for
UF1>]"
echo "         [-u2 <program for UF2>] [-o] [-s] [-h] [-u
<user/password>]"
echo "         <scale factor> <run_number>"
echo ""
echo "scale factor     : The scale factor of the run."
echo "update ||ism     : The parallelism to use for the UFs."
echo ""
echo "-p <program>     : Program for Query Stream."
echo "                 Default is $QPROG."
echo "-u1 <program>    : Program for UF1."
echo "                 Default is $U1PROG."
echo "-u2 <program>    : Program for UF2."
echo "                 Default is $U2PROG."
echo "-o               : Collect Oracle statistics."
echo "-s               : Collect System statistics."
echo "-u <user/passwd> : User/Password. Default is tpch/tpch."
echo "-h               : Displays this message."
}
set -- `getopt "p:u1:u2:osu:h" "$@"` || usage

while :
do
  case "$1" in
  -u1) shift; U1PROG=$1;;
  -u2) shift; U2PROG=$1;;
  -p) shift; QPROG=$1;;
  -o) OSTAT=1;;
  -s) SSTAT=1;;
  -h) usage; exit 0;;
  --) shift; break;;
  esac
  shift;
done


if [ "$#" -ne "3" ]
```


© 2004 Hewlett Packard Company. All rights reserved

```
then
  usage
  exit 1
fi

SF=$1
PARA=$2
RUN_ID=$3

OUT_DIR=${KIT_DIR}/audit/tests/${RUN_ID}
if [ ! -d $OUT_DIR ]
then
        mkdir $OUT_DIR
fi

TPCD_LOG=${OUT_DIR}
TPCD_RPT=${OUT_DIR}
OUT=${OUT_DIR}

let UF_SET="($PARA-1)*($NUM_STREAMS+1)+1"
START_SET=1
let STOP_SET=$NUM_STREAMS
let START_SET_UPDATE="($PARA-1)*($NUM_STREAMS+1)+2"
let
STOP_SET_UPDATE="$START_SET_UPDATE+$NUM_STREAMS
-1"

TPCD_LOG_FILE=${TPCD_LOG}/m${PARA}s0
TPCD_RPT_FILE=${TPCD_RPT}/m${PARA}s0inter
QRY_FILE=${TPCD_RPT}/qtemp.${PARA}s0
QUERY_PARAMETER=${TPCD_LOG}/qp${PARA}.0
SCRIPT_LOG_FILE=${TPCD_LOG}/m${PARA}timing
UF1_LOG=${TPCD_LOG}/m${PARA}s0rf1
UF2_LOG=${TPCD_LOG}/m${PARA}s0rf2
STREAM_COUNT_LOG=${TPCD_LOG}/m${PARA}tstrcnt

echo "TPC-H Test - RUN:${PARA} SEQUENCE:${RUN_ID} `date`"
> $SCRIPT_LOG_FILE
echo "TPC-H Test - RUN:${PARA} SEQUENCE:${RUN_ID} `date`"
> $TPCD_RPT_FILE
echo "Generates query template file with seed: `cat $SEED_FILE` for
stream 0" >> $SCRIPT_LOG_FILE
echo >> $SCRIPT_LOG_FILE

${QGEN} -c -r `cat $SEED_FILE` -p 0 -s ${SF} -l
$QUERY_PARAMETER > ${QRY_FILE}

START=`$GTIME`
echo "Start Power Test - RUN:${PARA} SEQUENCE:${RUN_ID}
Execution Starts $START, `date`" >>  $SCRIPT_LOG_FILE
echo "" >> $SCRIPT_LOG_FILE

# Execute UF1

SDATE=`date`
UF1_START=`$GTIME`
echo "Start UF1  $UF1_START, `date`" >> $SCRIPT_LOG_FILE

${UPD_SPT}/runuf1.sh ${UF_SET} >> $UF1_LOG 2>&1
# Execute Query Stream

UF1_END=`${GTIME}`
E1DATE=`date`

UF1_TIME=`echo $UF1_END - $UF1_START | bc`
echo UF1: Execution Time: $UF1_TIME >> ${TPCD_RPT_FILE}
echo Start Time: $UF1_START, $SDATE >> ${TPCD_RPT_FILE}
echo End Time: $UF1_END, $E1DATE >> ${TPCD_RPT_FILE}
echo "" >> ${TPCD_RPT_FILE}
echo "End UF1 $UF1_END, ${E1DATE}" >> $SCRIPT_LOG_FILE
echo UF1: Execution Time: $UF1_TIME >> $SCRIPT_LOG_FILE
echo >> $SCRIPT_LOG_FILE

echo "Start Query Part `$GTIME`, `date` " >> $SCRIPT_LOG_FILE

${QPROG} ${DATABASE_USER} q${QRY_FILE}
l${TPCD_LOG_FILE} r${TPCD_RPT_FILE}  > $DF  2>&1

# Execute UF2

UF2_START=`${GTIME}`
E2DATE=`date`

echo "End Query Part `$GTIME`, ${E2DATE}" >>
$SCRIPT_LOG_FILE
echo "" >> $SCRIPT_LOG_FILE

echo "Start UF2  $UF2_START, `date`" >> $SCRIPT_LOG_FILE
${UPD_SPT}/runuf2.sh ${UF_SET} >> $UF2_LOG 2>&1
UF2_END=`${GTIME}`
END=`${GTIME}`
EDATE=`date`

UF2_TIME=`echo $UF2_END - $UF2_START | bc`
echo UF2: Execution Time: $UF2_TIME >> ${TPCD_RPT_FILE}
echo Start Time: $UF2_START, $E2DATE >> ${TPCD_RPT_FILE}
echo End Time: $UF2_END, $EDATE >> ${TPCD_RPT_FILE}


echo "End UF2 $UF2_END, $EDATE" >>  $SCRIPT_LOG_FILE
echo UF2: Execution Time: $UF2_TIME >> $SCRIPT_LOG_FILE
echo >>  $SCRIPT_LOG_FILE

echo "End TPC-H Power Test - RUN:${PARA}
SEQUENCE:${RUN_ID}, $END, $EDATE" >> $SCRIPT_LOG_FILE
MEA_INT=`echo $END - $START | bc`
echo "Elapsed Time for TPC-H Power Test - RUN:${PARA}
SEQUENCE:${RUN_ID} is $MEA_INT" >> $SCRIPT_LOG_FILE
echo >> $SCRIPT_LOG_FILE

${KIT_DIR}/audit/abridge.pl ${TPCD_LOG_FILE}

i=$START_SET
PSEED=`cat $SEED_FILE`

while [ $i -le $STOP_SET ]; do
  TPCD_LOG_FILE=${TPCD_LOG}/mt${RUN_ID}_${i}.log
  TPCD_RPT_FILE=${TPCD_RPT}/mt${RUN_ID}_${i}.rpt
          QUERY_PARAMETER=${TPCD_LOG}/qp${PARA}.${i}
  QRY_FILE=${TPCD_RPT}/qtemp.${PARA}s${i}

  PSEED=`expr $PSEED + 1`
  ${QGEN} -c -r ${PSEED} -p ${i} -s ${SF} -l
$QUERY_PARAMETER > ${QRY_FILE}

  i=`expr $i + 1`
done

TH_START_D=`date`
TH_START_T=`${GTIME}`
echo >> $SCRIPT_LOG_FILE

rm -f /tmp/th_pipe1
mknod /tmp/th_pipe1 p
rm -f /tmp/th_pipe2
mknod /tmp/th_pipe2 p
i=$START_SET
```


© 2004 Hewlett Packard Company. All rights reserved

```
echo "Start Throughput Test - RUN:${PARA}
SEQUENCE:${RUN_ID} $TH_START_T, $TH_START_D"  >>
$SCRIPT_LOG_FILE

# starts a script to count the streams during the throughtput run
(scnt.sh $PARA $RUN_ID > $STREAM_COUNT_LOG &)

while [ $i -le $STOP_SET ]; do
  M_SDATE=`date`
  M_STIME=`${GTIME}`
            TPCD_LOG_FILE=${TPCD_LOG}/m${PARA}s${i}
            TPCD_RPT_FILE=${TPCD_RPT}/m${PARA}s${i}inter
  echo "Start Query Stream $i $M_STIME, ${M_SDATE}" >>
$SCRIPT_LOG_FILE
            QRY_FILE=${TPCD_RPT}/qtemp.${PARA}s${i}
  ${QPROG} ${DATABASE_USER} q${QRY_FILE}
l${TPCD_LOG_FILE} r${TPCD_RPT_FILE} | grep -v "Connected to
ORACLE" >> $SCRIPT_LOG_FILE &
  i=`expr $i + 1`
done

(${KIT_DIR}/audit/runTPCHus $RUN_ID $START_SET_UPDATE
$STOP_SET_UPDATE ${SF} $PARA >> $SCRIPT_LOG_FILE 2>&1
&)

wait
THQ_END_T=`$GTIME`
THQ_END_D=`date`
echo End all Query Streams $THQ_END_T, $THQ_END_D >>
$SCRIPT_LOG_FILE
print > /tmp/th_pipe1
read < /tmp/th_pipe2

TH_END_D=`date`
TH_END_T=`$GTIME`
echo End Update Stream ${TH_END_T}, ${TH_END_D} >>
$SCRIPT_LOG_FILE
echo >> $SCRIPT_LOG_FILE
echo "End Throughput Test ${TH_END_T}, ${TH_END_D}" >>
$SCRIPT_LOG_FILE
echo Execution Time Throughput Test: `echo ${TH_END_T} -
${TH_START_T} | bc` >> $SCRIPT_LOG_FILE

i=$START_SET
while [ $i -le $STOP_SET ]; do
            TPCD_LOG_FILE=${TPCD_LOG}/m${PARA}s${i}
  ${KIT_DIR}/audit/abridge.pl ${TPCD_LOG_FILE}
            i=`expr $i + 1`
done
PIDS=`ps -fu oracle | grep scnt.sh | grep -v grep | awk '{print $2}'`
kill -9 $PIDS
#calculate the metric
#analyze_streams.pl -f p -n $RUN_ID >
${TPCD_RPT}/tpch_metric.${RUN_ID}.${HID}.rpt
```

runTPCHus

```
#!/bin/ksh
. $KIT_DIR/env

SCRIPT_DIR=${KIT_DIR}/scripts
SQL_DIR=${KIT_DIR}/sql
UPD_DIR=${KIT_DIR}/update
UPD_SPT=${UPD_DIR}/scripts
SRC_DIR=${KIT_DIR}/utils
QRY_DIR=${KIT_DIR}/queries # this is the location of the query
template file
QGEN_DIR=${KIT_DIR}/dbgen
QGEN=${QGEN_DIR}/qgen
DSS_QUERY=${KIT_DIR}/queries
export DSS_QUERY

RUN_ID=$1
START_SET_UPDATE=$2
STOP_SET_UPDATE=$3
SF=$4
PARA=$5

OUT_DIR=${KIT_DIR}/audit/tests/${RUN_ID}
if [ ! -d $OUT_DIR ]
then
            mkdir $OUT_DIR
fi


TPCD_RPT=$OUT_DIR
SCRIPT_LOG_FILE=${OUT_DIR}/m${PARA}timing
OUT=$OUT_DIR

GTIME=${SRC_DIR}/gtime
HID=1

START=`$GTIME`
echo "Start Update Stream $START, `date`" >> $SCRIPT_LOG_FILE
echo "" >> $SCRIPT_LOG_FILE


#waiting for all the query streams to finish first
read < /tmp/th_pipe1

i=$START_SET_UPDATE
j=1
while [ $i -le $STOP_SET_UPDATE ]; do

  # Execute UF1

  UF1_LOG=${OUT_DIR}/m${PARA}s${j}rf1
  UF2_LOG=${OUT_DIR}/m${PARA}s${j}rf2
  RPT_FILE=${OUT_DIR}/m${PARA}s${j}inter

  SDATE=`date`
  UF1_START=`$GTIME`
            echo "Start UF1-${j} at ${UF1_START}, ${SDATE}" >>
${RPT_FILE}

  ${UPD_SPT}/runuf1.sh ${i} >> ${UF1_LOG} 2>&1
  UF1_END=`${GTIME}`
            EDATE=`date`
  echo "End UF1-${j} at ${UF1_END}, ${EDATE}" >> ${RPT_FILE}
            echo UF1-${j} Execution Time: `echo ${UF1_END} -
${UF1_START} | bc` >> ${RPT_FILE}

  # Execute UF2

  SDATE=`date`
  UF2_START=`${GTIME}`
  echo "Start UF2-${j} ${UF2_START}, ${SDATE}" >>
${RPT_FILE}

  ${UPD_SPT}/runuf2.sh ${i} >> ${UF2_LOG} 2>&1
  UF2_END=`${GTIME}`
  EDATE=`date`
  echo "End UF2-${j} at $UF2_END, ${EDATE}" >> ${RPT_FILE}
            echo UF2-${j} Execution Time: `echo ${UF2_END} -
${UF2_START} | bc` >> ${RPT_FILE}

  i=`expr $i + 1`
            j=`expr $j + 1`
done
```


© 2004 Hewlett Packard Company. All rights reserved

```
print > /tmp/th_pipe2
```

runuf1.sh

```
#!/bin/ksh
#
# $Header: runuf1.sh 25-oct-2001.15:56:04 mpoess Exp $
#
# runuf1.sh
#
# Copyright (c) 1999, 2001, Oracle Corporation.  All rights reserved.
#
#   NAME
#     runuf1.sh - <one-line expansion of the name>
#
# DESCRIPTION
#   runuf1.sh -l [<path name for reports>] -u [<uid/passwd>]
#           -p [<program>] <run_id> <scale factor> <pair number>
#           <parallelism>
# USAGE
#   To execute UF1.
#
#   NOTES
#     <other useful comments, qualifications, etc.>
#
#   MODIFIED   (MM/DD/YY)
#   mpoess     10/25/01 - change default directory for update sets
#   mpoess     10/17/01 - add support for external tables
#   mpoess     08/15/99 - Creation
#   mpoess     08/15/99 - Creation
#
#
. $KIT_DIR/env
O=${ORACLE_HOME}
UPDATE_DIR=${KIT_DIR}/update
SCRIPT_DIR=${UPDATE_DIR}/scripts
UTILS_DIR=${KIT_DIR}/utils
LOG_DIR=${UPDATE_DIR}/log
GTIME=${UTILS_DIR}/gtime
SF=${SCALE_FACTOR}
#PAR_HINT=${UPDATE_DOP}
PAR_HINT=96

LOGPATH=.
PASSWD=${DATABASE_USER}
if [ $# -lt 1 ];
then
  echo runuf1.sh setnum
  exit 1
fi
SETNUM=$1
i=1
PID=""

# perform the update function 1

START=`$GTIME`

# first create the temp tables

sqlplus /NOLOG << !

connect $PASSWD;
set timing on
set serveroutput on
set echo on

drop directory data_dir;
create directory data_dir as '/home/oracle/kit/update/scripts';
drop directory du1;
create directory du1 as '/home/oracle/dev/ff_97';
#create directory du1 as '/home/oracle/dev/ff_4/rf';
create table temp_l_et(
   l_orderkey          number ,
   l_partkey           number ,
   l_suppkey           number ,
   l_linenumber        number ,
   l_quantity          number ,
   l_extendedprice     number ,
   l_discount          number ,
   l_tax               number ,
   l_returnflag        char(1) ,
   l_linestatus        char(1) ,
   l_shipdate          date ,
   l_commitdate        date ,
   l_receiptdate       date ,
   l_shipinstruct      char(25) ,
   l_shipmode          char(10) ,
   l_comment           varchar(44)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
            records delimited by newline
            badfile 'l_et.${SETNUM}.bad'
            logfile 'l_et.${SETNUM}.log'
            fields terminated by '|'
            missing field values are null
      )
      location (
      du1:'lineitem.tbl.u${SETNUM}'
))
reject limit unlimited;

drop table temp_o_et;
create table temp_o_et(
   o_orderkey          number ,
   o_custkey           number ,
   o_orderstatus       char(1) ,
   o_totalprice        number ,
   o_orderdate         date ,
   o_orderpriority     char(15) ,
   o_clerk             char(15) ,
   o_shippriority      number ,
   o_comment           varchar(79)
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
            records delimited by newline
            badfile 'o_et.${SETNUM}.bad'
            logfile 'o_et.${SETNUM}.log'
            fields terminated by '|'
            missing field values are null
      )
      location (
      du1:'orders.tbl.u${SETNUM}'
))
reject limit unlimited;
alter table temp_l_et parallel ${PAR_HINT};
alter table temp_o_et parallel ${PAR_HINT};

alter session force parallel dml parallel (degree ${PAR_HINT});
alter session set isolation_level = serializable;
alter session set optimizer_index_cost_adj = 10;
```


© 2004 Hewlett Packard Company. All rights reserved

```
#alter session set events '8177 trace name errorstack level 1';
insert into orders (
  select
    o_orderdate          ,
    o_orderkey           ,
    o_custkey            ,
    o_orderpriority      ,
    o_shippriority       ,
    o_clerk              ,
    o_orderstatus        ,
    o_totalprice         ,
    o_comment
from temp_o_et);

insert into lineitem (
  select
    l_shipdate           ,
    l_orderkey           ,
    l_discount           ,
    l_extendedprice      ,
    l_suppkey            ,
    l_quantity           ,
    l_returnflag         ,
    l_partkey            ,
    l_linestatus         ,
    l_tax                ,
    l_commitdate         ,
    l_receiptdate        ,
    l_shipmode           ,
    l_linenumber         ,
    l_shipinstruct       ,
    l_comment
from temp_l_et);
commit;

drop table temp_l_et;
drop table temp_o_et;

exit;
!

END=`$GTIME`

# Done

echo ""
echo "Update Function 1 Set $SETNUM done!"
echo "Elapsed Time is `echo $END - $START | bc`"
echo ""
```

runuf2.sh

```
#!/bin/ksh
#
# $Header: runuf2.sh 25-oct-2001.15:56:05 mpoess Exp $
#
# runuf2.sh
#
# Copyright (c) 1999, 2001, Oracle Corporation.  All rights reserved.
#
#    NAME
#      runuf2.sh - <one-line expansion of the name>
#
# DESCRIPTION
#    runuf2.sh [-u <uid/passwd to login>] [-p <program>] <run_id>
#              <scale factor> <pair number> <parallelism>
# USAGE
#    To execute UF2.
#
#    NOTES
#      <other useful comments, qualifications, etc.>
#
#    MODIFIED   (MM/DD/YY)
#    mpoess      10/25/01 - change default directory for update sets
#    mpoess      10/17/01 - add support for external tables
#    mpoess      08/15/99 - Creation
#    mpoess      08/15/99 - Creation
#
. $KIT_DIR/env
UPDATE_DIR=${KIT_DIR}/update
SCRIPT_DIR=${UPDATE_DIR}/scripts
UTILS_DIR=${KIT_DIR}/utils
GTIME=${UTILS_DIR}/gtime
LOG_DIR=${UPDATE_DIR}/log
#PAR_HINT=${UPDATE_DOP}
PAR_HINT=96
SF=${SCALE_FACTOR}
PASSWD=${DATABASE_USER}

if [ $# -lt 1 ]
then
  usage
  exit 1
fi

SETNUM=$1

i=1
PID=""

START=`$GTIME`
# first create the temp tables

sqlplus /NOLOG << !

connect $PASSWD;
set timing  on
set serveroutput on
set echo on

drop directory du1;
create directory du1 as '/home/oracle/dev/ff_97';
#create directory du1 as '/home/oracle/dev/ff_4/rf';
drop table temp_okey_et;
drop table temp_okey;

create table temp_okey_et(
   t_orderkey          number
)
organization external (
type ORACLE_LOADER
default directory data_dir
access parameters
(
            records delimited by newline
            badfile 'okey.${SETNUM}.bad'
            logfile 'okey.${SETNUM}.log'
            fields terminated by '|'
            missing field values are null
      )
      location (
      du1:'delete.${SETNUM}'))
reject limit unlimited;

alter table temp_okey_et parallel ${PAR_HINT};
--alter table temp_okey_et parallel 96;

create table temp_okey parallel ${PAR_HINT} nologging as select *
from temp_okey_et;
```


© 2004 Hewlett Packard Company. All rights reserved

```
--create table temp_okey parallel 96 nologging as select * from
temp_okey_et;

create unique index i_temp_okey on temp_okey (t_orderkey) parallel
${PAR_HINT} nologging compute statistics;
--create unique index i_temp_okey on temp_okey (t_orderkey) parallel
96 nologging compute statistics;

analyze table temp_okey estimate statistics sample 2 percent;

alter session  force parallel dml parallel ${PAR_HINT};
alter session set isolation_level=serializable;
alter session set optimizer_index_cost_adj = 10;

delete from (select /*+ use_nl(o)  */ o.rowid from orders o, temp_okey t
where o.o_orderkey = t.t_orderkey order by 1);

delete from (select /*+ use_nl(l) */ l.rowid from lineitem l,temp_okey t
where l.l_orderkey = t.t_orderkey order by 1);

commit;

drop table temp_okey;
drop table temp_okey_et;
exit;
!

END=`$GTIME`

# Done

echo ""
echo "Update Function 2 Set $SETNUM done!"
echo "Elapsed Time is `echo $END - $START | bc`"
echo ""
```


© 2004 Hewlett Packard Company. All rights reserved

Appendix G: Price Quotes

© 2004 Hewlett Packard Company. All rights reserved

-----Original Message-----
From: MaryBeth Pierantoni [mailto:mary.beth.pierantoni@oracle.com]
Sent: Friday, June 11, 2004 5:38 PM
To: Raghunath Othayoth
Subject: Pricing

Product	**Price ($USD)**	**Quantity**	**Extended Price**
Oracle Database 10g Enterprise Edition for 3 years, Named User Plus	$10,000	48	$480,000
Real Application Clusters for 3 years, Named User Plus	$5,000	48	$240,000
Partitioning for 3 years, Named User Plus	$2,500	48	$120,000
Database Server Support Package (12 servers, 3 years)	$24,000	3 years	$72,000
Oracle Mandatory E-Business Discount			<$182,400>
TOTAL			$729,600

Contact: MaryBeth Pierantoni, mary.beth.pierantoni@oracle.com, 916-315-5081

InfiniCon Price Quote

ALL INFORMATION CONTAINED ON THIS DOCUMENT IS CONFIDENTIAL TO THE INTENDED RECIPIENT

From:	To:	
InfiniCon Systems	Raghunath Othayoth	Date 6/10/2004
680 American Avenue, Suite 100	Hewlett-Packard Corporation	Sales Rep Ed Bollet
King of Prussia, PA 19406	20555 State Highway 249	Sales Order #
		Ship Via
	Houston, Tx 77070	Payment Tern 30 Days
Contact: Ed Bollet, 503-819-0897		
Or: Ed Bollet, 503-819-0897		P.O. #

DESCRIPTION: 12 Node 10Gig FBB Fabric for Oracle Benchmarks
DISCOUNT TIER: End User ASP

Product	Part #	QTY	Description				End User Price	Annual Maintenance
Switch	3016-SWI	1	InfinIO 3016 - 32 Port (16 X 16 Clos) InfiniBand switch				$8,769	$915
HCA	7104-HCA-128LP	12	InfiniServ 7000 Low Profile HCA 128 MB Std				$8,487	$450
	Total Server Cables	0						
Cables								
	7104-3M-Cable	0	4x - 4x cable - 3 meters					
	7104-5M-Cable	12	4x - 4x cable - 5 meters				$1,546	
Subnet Mgr.	SW79-SM-V7K	0	Embeded InfinIO 7000 InfiniView Fabric Manager					
	SW79-SM-V3K	0	Embeded InfinIO 3000 InfiniView Fabric Manager					
	SW79-SM-VXT	0	Server-based InfiniViewFabric Manager					
	SW79-SM-V2K	1	**Embeded InfinIO 2000 InfiniView Fabric Manager**				$1,000	$525
			Sub Total				**$19,802**	$1,890
			Per Port Price				**$1,650**	
Documentation		1	**Comprehensive Product Documentation**				INCLUDED	
			Getting Started Installation Guide					
			Capabilities Manual					
			Users Guide					
			Onsite				OPTIONAL	
		2	**Yrs Additional Maintenance**				**$3,780**	
			Grand Total				**$23,582**	

Warranty HW 12 Months
Support HW 12 Months
SW: 12 Months with 15% add-on to SW List

June 9, 2004

Raghunath K. Othayoth
ISS - Solutions and Strategy
Hewlett-Packard Company
281-518-2748 tel

Per your request for information on Red Hat Enterprise Linux pricing, please find the quote below. These prices are valid for 30 days.

Part Number	***Description***	***Unit Price***	***Quantity***	***Price***
SVC0084	Red Hat Enterprise Linux AS 3.0 (Premium Edition for x86) 3-year 24x7 support subscription	$2,499	12x3	$89,964
TOTAL				$89,964

Products are orderable through www.redhat.com or Red Hat Sales 1-888-REDHAT-1. If we can be of any further assistance, please contact Gerry Riveros at griveros@redhat.com.

*Support and maintenance for software includes annual phone and web support and continuous proactive update and upgrade support via Red Hat Network.